

**G. Bäşimowa, B. Iwanow,
O. Atdaýewa, B. Orazow**

AWTOMATLAŞDYRYLAN ULGAMLARYŇ WE KOMPÝUTER TEHNIKASYNYŇ PROGRAMMA ÜPJÜNÇILIGI

I kitap

Ýokary okuw mekdepleri üçin okuw kitaby

*Türkmenistanyň Bilim ministrligi
tarapyndan hödürlenildi*

Aşgabat
Türkmen döwlet neşirýat gullugy
2016

Bäşimowa G. we başg.

B Awtomatlaşdyrylan ulgamlaryň we kompýuter tehnikasynyň programma üpjünçiligi. Ýokary okuw mekdepleri üçin okuw kitaby. – A.: Türkmen döwlet neşirýat gullugy, 2016.

Dünýäniň innowasion tehnologiýasynyň çylşyrymly gurluşyny, dolandyrylyşyny, sazlaşykly döremegini, özboluşly aýratynlyklaryny we mümkinçiliklerini öwrenmek we olary dolandyrmak diňe programma üpjünçiliklerine baglydyr.

Şonuň üçin hem häzirki döwürde duş gelýän meseleleriň oňyn çözülmegi, esasan, onuň bilen bagly bolan maglumatlary öz wagtynda toplamak, döretmek we olary dolandyrmak bilen baglanyşyklydyr. Bu işiň wagtynda we üstünlikli çözülmeginde häzirki zaman programmirlleme dilleriniň we ulgamlaryň ähmiýeti hem uludyr.

Bu okuw kitaby «Awtomatlaşdyrylan ulgamlaryň hem-de kompýuter tehnikasynyň programma üpjünçiligi» hünäri boýunça bilim alýan talyplar üçin niýetlenip, olaryň ýurdumyzyň ykdysady pudaklarynyň elektron resminama dolanyşygynyň we gündelik iş dolandyrylyşynyň çylşyrymly meselelerini tiz çözmekde giň gözýetimli programmist-ykdysatçy hünärmenlere gollanma bolup hyzmat eder.



**TÜRKMENISTANYŇ PREZIDENTI
GURBANGULY BERDIMUHAMEDOW**



TÜRKMENISTANYŇ DÖWLET TUGRASY



TÜRKMENISTANYŇ DÖWLET BAÝDAGY

TÜRKMENISTANYŇ DÖWLET SENASY

Janym gurban saňa, erkana ýurdum,
Mert pederleň ruhy bardyr köňülde.
Bitarap, garaşsyz topragyň nurdur,
Baýdagyň belentdir dünýäň önünde.

Gaytalama:

Halkyň guran Baky beýik binasy,
Berkarar döwletim, jigerim-janym.
Başlaryň täji sen, diller senasy,
Dünýä dursun, sen dur, Türkmenistanym!

Gardaşdyr tireler, amandyr iller,
Owal-ahyr birdir biziň ganymyz.
Harasatlar almaz, syndyrmaz siller,
Nesiller döş gerip gorar şanymyz.

Gaytalama:

Halkyň guran Baky beýik binasy,
Berkarar döwletim, jigerim-janym.
Başlaryň täji sen, diller senasy,
Dünýä dursun, sen dur, Türkmenistanym!

GIRIŞ

Kompýuter ylmylary, hususan-da programmirlleme inženeriýasy bilim ulgamynyň iň meşgur we çalt ösýän ugurlarydyr. Muňa öňdebaryjy ýurtlarda ilatyň maglumat ulgamynda iş üpjünçiliginiň 60%-ini düzýändigini şaýatlyk edýär. Hut şonuň üçin «Kompýuter ylmylary we maglumat tehnologiýalar» ugrundaky hünärler iň abraýly hünär hasaplanýar.

Bu gün maglumat örän wajyp strategiki resursdyr. Kompýuterleriň programma üpjünçiligi – häzirki zaman maglumat ulgamlarynyň we tehnologiýalarynyň esasy komponentidir. Olaryň işlenip taýýarlanylmagyna örän uly maliýe, tehniki we zahmet resurslary gönükdirýäler.

XX asyryň 90-njy ýyllaryndan bäri kompýuter programma üpjünçiliginiň hilini kämilleşdirmek esasy we wajyp meseleleriň hataryna girdi. Kompýuteriň mümkinçilikleri bitewüligine programma üpjünçiliginiň mümkinçilikleri bilen kesgitlenýär. Häzirki zaman kompýuteriň mümkinçilikleri örän uludyr we islendik meseläni çözmeklik üçin apparat üpjünçiligine bolan çäklendirmeler indi aradan aýryldy. Emma programma üpjünçiligine bolan çäklendirmeler henizem wajypdyr. Häzirki wagtda wajyp meseleleriň hataryna indikileri goşmak gerek:

- apparat üpjünçiligiň çylşyrymlylygy biziň bu apparat üpjünçiliginiň potensial mümkinçiliklerini ulanyp bilýän programma üpjünçiligini düzmek mümkinçiligimizden örän öňe gidýär;
- biziň täze programmalary döredip bilijilik mümkinçiligimiz täze programmalara edilýän talaplaryndan yza galýar;
- biziň bar bolan programmalarymyzy işletmek mümkinçiliklerimize olaryň işlenilip düzülişiniň hiliniň pesligidir.

Bu meseleleri çözmek için programma üpjünçiligini döretmek prosessini sowatly gurap bilmek, şeýle hem programma ulgamlary döretmekligiň tehnologiiki düzgünlerini amala aşyrmak zerur.

Okuw kitabynda programmirlmekligiň usulyýetleri, çemeleşmeleri we tehnologiýalary giňişleýin görkezilýär, olaryň klassifikasiýalary getirilýär. Şeýle hem kitapda awtomatlaşdyrylan ulgamlaryň we kompýuter tehnikasynyň programma üpjünçiligini işläp taýýarlamakda ulanylýan düzgünler, modeller we usullar hakda yzygider düşünje berilýär. Programma taslamasynyň dolandyrylyşy we onuň aýratynlyklary hem giňişleýin düşündirilýär.

Bu kitap «Awtomatlaşdyrylan ulgamlaryň we kompýuter tehnikasynyň programma üpjünçiligi» hünärinde okaýan talyplar hem-de programma üpjünçiligini işläp düzüjiler we programma taslamalaryň menejerleri üçin niýetlenendir.



I. PROGRAMMIRLEMEĞİN ESASY DÜŞÜNJELERİ

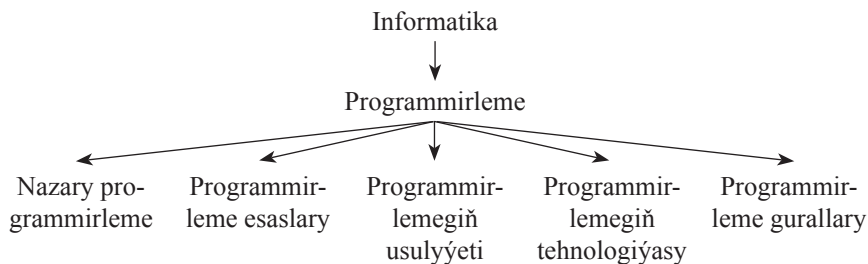


1. Programmirleme we onuň bölümleri

1.1. Programmirleme – informatikanyň bölegi we onuň bölümleri hökmünde

Programmirleme – bu informatikanyň maglumatlary işläp taýýarlamak prosessleriniň ugruny öwrenýän bölegidir. Programmirlemede indiki bölümler anyk tapawutlandyrylýar (*1.1-nji surat*):

- *nazary programmirleme*. Onuň öwrenmek obýekti bolup, programmalaryň matematiki abstraksiýalary çykyş edýär. Bu ugruň başyny başlan alymlardan ilkinjisi A.A. Lýapunow. Bu alym programmirleme ugry boýunça özbaşdak nazaryýetini döretmekligi talap edýän bilimiň aýratyn bölegini gördi. Lýapunow ilkinji bolup, programma ýazmaklykda operatorly usuly ulanmaklygy teklipti. Hut şol usulda hem ilkinji gezek programma düşünjesiniň resmi formasy kabul edilip, programmanyň ekwiwalent özgertmelerini işlemeklik meselesi goýuldy. Ekwiwalent özgertmelerini amala aşyryp bolýan obýektleri hökmünde, programmanyň dolandyryjy strukturasyny saklaýan, şeýle hem operatorlaryň we logiki şertleriň jikme-jik beýan etmekden ünsi çekýän programma shemalar girizildi. Programma shemalarynyň nazaryýetiniň aýratyn ugruny programmalaryň algebraiki modelleriniň nazaryýeti emele getirdi;
- *programmirleme esaslary*. Programmirlemek tejribesi üçin fundamental esaslary hökmünde wajyp bolan konsepsiýalar we endikler aşakdakylardan ybaratdyr:
 - maglumatlaryň strukturalary we olaryň kompýuteriň ýadyndaky görnüşi;
 - maglumat gözlegi, çyzykly massiwleri we faýllary tertiplemek;



1.1-nji surat. Informatikanyň programmirlleme bilen baglanyşygy

- formal diller we beýleki abstrakt maşynlar;
- programmalaryň sintaksiz seljermegi;
- algoritmleriň çylşyrymlylyk nazaryýeti;
- meseleleriň spesifikasiýasy, programmalaryň häsiýetlerini subut etmeklik, programmalaryň awtomatiki sintezi;
- programmirlleme dilleriniň ýygnamy (semantikasi).
- *programmirllemegiň usulyýeti (metodologiýasy)*. Gurmaklygyň esaslarynyň nukdaý nazaryndan usullary öwrenýär. Bu programma üpjünçiligini işläp taýýarlamak prosesinde ulanylyan, umumy filosofiki çemeleşiş bilen birleşdirilen usullaryň jemidir.
- *programmirllemegiň tehnologiýasy* – tehnologiki prosesleri we olaryň bolup geçme tertibini (usullary, bilimleri we serişdeleri ulanmak bilen) öwrenýär.
Tehnologiki proses – bu her biri käbir tebigy proseslere we adam işine esaslanan işleriň (tehnologiki prosesleriň) berlen obýekti döretmeklige esaslanan yzygiderliligi. Usullar, bilimler we serişdeler dürli proseslerde, diýmek, tehnologiýalarda ulanylyp bilner.
- *programmirlleme gurallary* – programmirlleme prosesini goldaýan ähli gurallar.

1.2. Programmirllemegiň inženeriýasy

XX asyryň 60-njy ýyllarynyň ahyrynda akademik A.P. Ýerşow «Software Engineering» adalgasyny «Programmirleme tehnologiýasy» diýip atlandyrdy. Emma bu adalga bilen bir hatarda

«Programmirlemegiň inženeriýasy» adalgasy hem giňden ulanylyp, ýöne, köplenç, bu adalgalar şol bir zady aňladýarlar. Bu adalgalaryň tapawudy barada aýtsak, onda programmirleme inženeriýasy informatikanyň has giň bölegini tutýar we programma üpjünçiligini döretmekligiň tejribe meseleleriniň ähli spektrini öz içine alýar. Programmirleme inženeriýanyň esasy ideýasy – programma üpjünçiligini döretmek prosesiniň formal proses hasaplanyp, ony öwrenip kämilleşdirip bolýanlygyndadyr. Bu bölüm 1968-nji ýylda «Programma üpjünçiliginiň krizisiniň» netijesinde emele geldi. Frederik Bruks tarapyndan programmirleme inženeriýasynyň häsiýetli meseleleri we wezipeleri häzirki güne çenli hem wajypdyr [Frederick Brooks 1999]. Olaryň içinde indikileri mysal getirmek bolar:

- ulgamlary emele getirýän programmalary nädip döretmeli we taslamaly?
- ygtybarly, sazlanan, resmileşdirilen hem-de ugradyjy önüm bolup durýan programmalary, ulgamlary nädip taslamaly we döretmeli?
- örän çylşyrymlylyk şertlerde intellektual barlagy nädip amala aşyrmaly?

«Programmirlemegiň inženeriýasy» bölümüniň şu günki mazmuny örän dinamiki we ylmy ugurlaryň köp sanyny öz içine alýar. Olaryň programmirleme inženeriýa boýunça halkara forumlarda özara alyp maslahatlaşylýanlarynyň käbirlerini getireliň:

- programma üpjünçiligini işläp taýýarlamak prosesi;
- programma önümine talaplary döretmek;
- ters (rewersiwn) inženeriýa – kod boýunça spesifikasiýany dikeltmek prosesi;
- testirlemek;
- programma üpjünçiliginiň ewolýusiýasy;
- programma üpjünçiliginiň arhitekturasy;
- obýekte gönükdirilen programmirleme;
- programmirlemegiň inženeriýasy we «aralyk» (tor operasiýon ulgamy bilen ulanyjynyň programmasynyň arasynda) programma üpjünçiligi;
- programma üpjünçiliginiň seljermesi;
- formal spesifikasiýalar;

- programmirlemegiň inženeriýasynyň matematiki esaslandyrmalary;
- programmirlemegiň inženeriýasy we programma önüminiň önümçiligi;
- hakyky wagt ulgamlary üçin programmirlemegiň inženeriýasy;
- mobil ulgamlar üçin programmirlemegiň inženeriýasy;
- programmirlemegiň inženeriýasy we Internet;
- programma üpjünçiliginiň ykdysadyýeti;
- programma üpjünçiliginiň metrikalary;
- programmirlemegiň inženeriýasy okuw dersi hökmünde.



2. Programmirlemegiň esasy düşüňjeleri we kesgitlemeleri

2.1. Programma we programma önümi

Ilki bilen kitabyň dowamynda ulanyljak kesgitlemeleri we düşüňjeleri girizeliň. Indiki iki kesgitlemeleriň esasynda Frederik Bruksyň düşüňjeleri ýatyr. Olar programma bilen programma önümiň arasyndaky tapawudy görkezýärler. Programma «özün üçin» döredilýär, programma önümi bolsa «Buýrujy üçin» döredilýär we onuň döredilişine has çynlakaý çemeleşmeklik göz önünde tutulýar.

Programma (program). Bu onuň döredilen ulgamynda goýberilmegi üçin bolan gutarnykly önüm. Anyk kesgitlenen interfeýsler, funksiýalar bilen formatlar boýunça ylalaşylan, şeýle hem uly meseleleri çözmek üçin bilelikde doly önümi emele getirýän we özara baglanyşykly programmalaryň jemi *programma toplumyny (kompleksini)* emele getirýär.

Programma önümi (program product). Bellibir talaplary kanagatlandyryň we satmak ýa-da beýleki adamlara ulanmaga bermek üçin niýetlenilen programma ýa-da programma toplumu. Programma önümi programmirlemegiň bir umumy stiline laýyk bolmaly, şeýle hem testden gowy geçirilip, ulanyş we sazlanýş düzgünlerini beýan edýän resminamalar bilen bilelikde berilmeli.

Umuman aýtsak, programma we programma önümi barada kesgitlemeler we düşüňjeler dürli awtorlar tarapyndan dürlüçe berlip bil-



ner. Indiki iki sany düşünje programmanyň beýleki tarapyny – onuň hasaplaýyş ulgamynyň fiziki böleklerinden (apparat serişdelerinden we aparat üpjünçiliginden) tapawudyny görkezýärler.

Programma serişdesi (software). Programmany ulanmaklyk boýunça resminamalary bilen üpjün edilen, maglumat göterijilerdäki programma ýa-da programma toplumy [Žogolew 1996].

Programma üpjünçiligi (software). Hasaplaýyş ulgamyň düzümine girýän programma serişdeleriniň toplumy. Bu serişdeleriň maksady – kompýuteriň işiniň netijeliligini üpjün etmek we ulanyja hyzmatlaryň käbir görnüşlerini hödürlemek. Programma üpjünçiliginiň iki sany esasy görnüşini tapawutlandyrýarlar;

- ulgamlaryn programma üpjünçiligi. Bu programma üpjünçiligi aparat üpjünçiliginiň üstüni ýetiriji bolup, onuň esasy komponenti – operasion ulgamy bolup durýar.
- amaly programma üpjünçiligi. Bu programma üpjünçiligi spesifikasiýa kompýuteriň orny we ulanyjynyň talaby boýunça kesgitlenýän amaly programmalar ybarat. Mysallary hökmünde amaly programmalar toplumlaryny we maglumatlar bazalaryny dolandyryan ulgamlaryny görkezmek bolar.

2.2. Algoritmeler

Algoritm – bu ýerine ýetirip bolýan elementar işleriň yzygiderligini görkezýän şol ýa-da beýleki umumy usulyň anyk beýany.

Kompýuter – bu hasaplaýjy, ol programma düşünmeýär ol diňe programmany ýerine ýetirýär. Kompýutere programmanyň ýerine ýetirilişiniň gidişini görkezmek üçin iň adaty usul – ony algoritm görnüşde (algoritm dilinde) ýazmak. Häzirki döwürde «Algoritm» sözi köp ýerlerde resept, proses, usul ýaly sözlere analogiki bolup durýar. Algoritm 5 sany wajyp häsiýetlere eýedir:

- *gutarnyklylyk*. Ädimleriň gutarnykly sany ýerine ýetirilenden soň algoritm hökman gutarmaly;
- *kesgitlilik (determinirlilik)*. Algoritmiň her bir ädimi anyk kesgitli bolmaly;
- *giriş maglumatlaryň bolmaklygy*. Algoritmiň işe başlamazyndan öň, onuň giriş maglumatlaryň birnäçesi bar ýa-da bu

giriş maglumatlaryň sany onuň ýerine ýetirilişiniň dowamynda dinamiki ýagdaýda kesgitlenýär;

- *çykyş maglumatlaryň bolmaklygy*. Algoritmiň giriş maglumatlary bilen bellibir derejede baglanyşykly bir ýa-da birnäçe çykyş maglumatlary bar;
- *netijelilik*. Algoritm, eger-de onuň operatorlary, olary gutarnykly wagt aralygynyň dowamynda ýerine ýetirmek üçin ýeterlik derejede ýönekeý bolsalar netijeli diýip hasaplanýar.

Algoritm bilen ylymlaryň indiki ugurlary bagly:

- *algoritmeleriň seljerilişi*. Bu bölümiň predmeti berlen algoritm üçin işjeň häsiýetlerini kesgitlemekden ybarat. Mysal üçin, köplenç, algoritmiň çalt işlenmegi talap edilýär;
- *algoritmeleriň nazaryýeti*. Bu bölüme bellibir ululyklary hasaplamaklygyň netijeli algoritmeleriniň barlygy ýa-da ýoklugy baradaky meseleler degişli;
- *algoritmeleriň gurluşy*. Bu bölümde algoritmeler ýazylanda, ulanylýan standart usullara we ýollara seredilýär.

2.3. Modeller we modelirleme

Model – bu hakykylygyň ýenilleşdirilen düşündirilişi. Model bolup, mysal üçin ulgamyň çyzygysy çykyş edýär. Model – bu ulgamyň semantiki ýapyk abstraksiýasy – modeliň esasy häsiýeti. Model işleniljek ulgama has gowy düşünmek, ony wizualizirmek, onuň düzümini we özüni alyp barşyny kesgitlemek üçin gurulýar. Çylşyrymly ulgamlary modelirlemek hökman, bolmasa biz olary bir bitewi hökmünde kabul edip bilmeris. Dürli modeller ulgamyň gurluşyny dürli görkezýärler, şonuň üçin hem olaryň hiç biri-de ony kesgitlemeýär.

Modelirleme – ulgamyň öwrenilýän häsiýetlerini beýleki görnüşli obýektlere geçirmeklige esaslanan ulgamlary öwrenmek usuly. Modelirlemekde esasy bolup modelirlenýän obýekt bilen onuň modeliniň arasyndaky meňzeşlik gatnaşygy çykyş edýär. Bu biziň daşymyzdaky hakykylygy öwrenmekligiň esasy usullarynyň biri. Modelleri taýýarlamaklygyň inzener usulyýeti ähli ýerde kabul edilen we saklanan hasap edilýär hem-de birnäçe wajyp meseleleri çözmeklige mümkinçilik berýär:

- ulgamy wizualizirmek;
- ulgamyň gurluşyny we onuň özüni alyp barşyny kesgitlemek;
- kabul edilýän çözümleri resmileşdirmek.

Galyberse-de modelirmek – bu çylşyrymlylyk meselesini çözmeklige synanyşyk. Modelirmegiň dört esasy prinsipi bar:

- modeli saýlap almaklyk meseläni çözmeklige çemeleşmäge we bu çözüdiň görnüşini kesgitleýji täsir edýär;
- her bir model abstraksiýanyň dürli derejeleri bilen amala aşyrylyp bilner. Şunlukda, abstraksiýanyň bir derejesinden beýlekä geçmeklik bize täze modeli berýär;
- hakykylyga has ýakyn modeller (käbir bellenen kriteriler boýunça) iň gowy hasap edilýärler;
- birnäçe modelleriň jemini ulanmak gerek.

Modelleriň, olaryň esasynda baglylykda üç sany: natural modeller (mysal üçin, maketler), göz önüne getirilýän modeller (mysal üçin, geometriki nokat) we maglumat modeller (mysal üçin, programma) görnüşini tapawutlandyryrlar. Maglumat modelleri hem öz gezeginde üç görnüşe emele getirýärler:

- *diskriptiw modeller*. Olar beýan etmek dilinde aňladylyrlar. Bu adaty ýa-da ýörite: ylmy (algoritmeler üçin) we tehniki (programmalar üçin) diller bolup biler;
- *görkezme modeller*. Olar görkezme dilinde aňladylyrlar. Mysal üçin, suratlar, çyzgylar ýa-da fotosuratlar;
- *garyşyk modeller*. Olar beýan ediş dilini hem görkezme dilini hem ulanýrlar. Mysal üçin, diagrammalar, shemalar, tablislalar, graflar.

2.4. Programmanyň ýaşayyş sikli

Programmanyň ýaşayyş sikli – muňa programmany döretmekden başlap, ony ulanmak görnüşleriniň ählisiniň bes edilmegine çenli döwrüniň ählisi girýär. Diýmek, bu döwür, programmanyň işlenip taýýarlanylş we ulanylyş tapgyrlarynyň ählisini öz içine alýar. Ýaşayyş sikliniň ýönekeý düşüňjesi indiki:

- seljermek;
- taslamak;

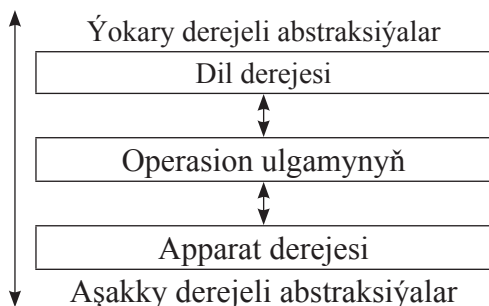
- programmirmek;
- testirmek we düzetmek;
- alyp barmak we ulanmak döwür.

2.5. Platformalar

Öňden belli bolşy ýaly, ýönekeý ýagdaýda kompýuteri guramagyň üç derejeli arhitekturasy apparat, amallar we dil derejelerinden düzülen (2.1-nji surat).

Platforma – bu programmany düzüjä görünýän programma önümini goldaýan serişdeler. Platformalaryň indiki görnüşleri kesgitlenen:

- *apparat platformasy* – programma önümini ornaşdyryp boljak apparat arhitekturasynyň görnüşi. Intel, HP Power prosessorynyň toparlaryndaky (maşgalalaryndaky) platformalar has giňden ýaýran;
- *amallar platformasy* – programma önüminiň işläp biljek operasion ulgamynyň we ulgam interfeýsiniň görnüşi. Has meşgul bolan operasion ulgamlary: MS-DOS, Windows we Unix. Tor platformasy adaty ýagdaýda amallar platformasynyň düzümine girýär we amallar platformasynyň tor interfeýsini emele getirýär. Amallar platformasynyň bir bölegi bolup, operasion ulgamy bilen berilýän kitaphanalar çykyş edýärler we olar bilen işlemek üçin amaly programma düzüjiniň interfeýsi ulanylýar;
- *dil platformasy* – programmirmek dili we programma önümi amala aşyrylyp biljek amaly programmirmekigiň interfeýsi (kitaphanalar). Has meşgul dil platformalary: C, C++, Java.



2.1-nji surat. Kompýuteriň üç derejeli arhitekturasy

Programma önümini ulanyjynyň, düzgün bolşy ýaly, önümiň haýsy apparat we amallar platformalar üçin niýetlenendigini bilmelidigini hem bellemek gerek.

Amaly programma düzüjiniň interfeýsine (kitaphanalaryna) amallar platformasy-da, gurallar platformasy-da, dil platformasy-da eýe bolup bilerler. Ondan başga-da, şol bir platforma amaly programma düzüjiniň interfeýsi boýunça mümkinçilikleriň dürli toplumlaryna eýe bolan dürli neşirlerde (redaksiýalarda) çykarylýp bilner. Mysal üçin, Java platformasy üçin şeýle neşirler çykarylýar:

- *korporatiw (enterprise) neşir* – kärhana derejesindäki we merkezi serwer bilen dolandyrylýan köp derejeli goşundylary döretmek we ösdürmek üçin niýetlenen;
- *standart (standard) neşir* – platforma ara uniwersal goşundylary işläp taýýarlamak, dolandyrmak we ösdürmek üçin niýetlenen;
- *mikro (micro) neşir* – smart-kartlardan başlap hasaplaýjy enjamlara çenli goşundylary işläp düzmek, ösdürmek we dolandyrmak maksady bilen kiçi awtonom ulanyjylar üçin niýetlenen.



Taslamalary işläp taýýarlamak prosesi

3.1. Taslamalar we olaryň häsiýetleri

Taslama (project) – bu programma önümine gönükdirilen programmany işläp düzüjileriň işleriniň birleşmesi. Programmany işläp düzüjileriň işleri aýratyn bir prosese (taslamak, testirmek we ş.m.) däl-de, hut önümiň özüne gönükdirilendigini aýratyn bellemek gerek. İşläp taýýarlamalaryň üç sany esasy kategoriýalaryna, olara degişli taslamalara seredeliň we kompilyatory işläp taýýarlamaklyk ugrundan mysallar bilen berkideliň:

- programma önümiň *häsiýeti (feature)* – önümiň käbir spesifiki aspektine gönükdirilen arhitektura elementi. Häsiýet bir ýa-da birnäçe komponentde amala aşyrylyp bilner. Häsiýetiň mysaly hökmünde kompilyatoryň dile bagly bölümünde sazlaýjy üçin, maglumaty generirleýji ulgam bolup biler;



- *komponent (component)* – häsiýetleriň toplumy. Adaty ýagdaý-da komponent möhümliligini göz önünde tutýar we komponenti işläp taýýarlaýjylar oňa jogap berýärler. Komponentleriň mysallary – Pascal we FORTRAN dilleriniň kompilýatorlarynyň dile bagly bolan bölegi;
- *önüm (product)* – ulanyjynyň bellibir meselelerini çözmek maksady bilen birleşdirilen komponentleriň jemi. Önümiň üstünde önümi işläp düzüjileriň topary işleýär. Önümiň görnüşli mysallary – Paskal-kompilýatory ýa-da FORTRAN-kompilýatory. Taslamanyň netijesi bolup kompilýatoryň bellibir görnüşü (mysal üçin, 2.0) çykyş eder.

Taslamalaryň ýene-de käbir häsiýetlerine seredeliň.

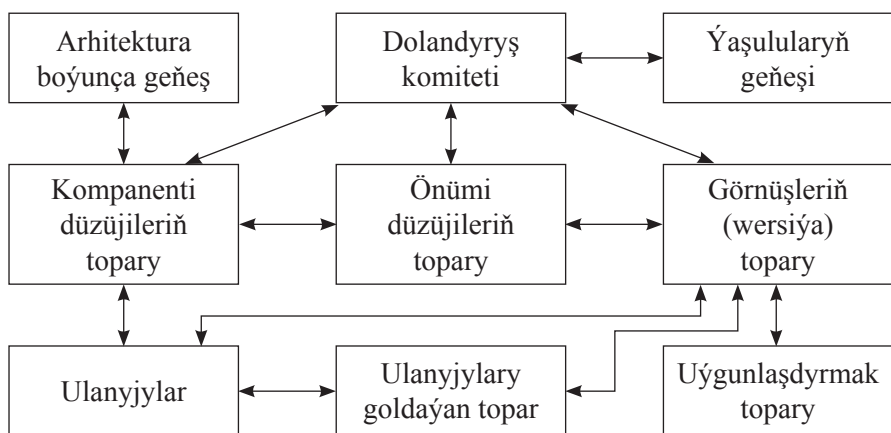
- taslamanyň buýrujynyň görnüşü bilen kesgitlenen *spesifikasy* bar. İşläp taýýarlamak prosesinde has düýpli tapawut nyşanlary bolup, dolandyryş, döreýiş we pikiri barlamak prosesslerinde tapawut nyşanlary çykyş ederler. Taslama bellibir diapazonda bolup biler, bu diapazonyň çäk bahalary bolup indikiler çykyş ederler:
 - ulanyjylaryň giň topary üçin taslama;
 - bellibir buýrujy üçin taslama.
- Taslamanyň ýene-de bir häsiýeti bolup onuň gerimliligi (masştablylygy) çykyş edýär. Taslamanyň gerimine baglylyk-da, onuň taslamalarynyň dört kategoriýasyny bellemek bolar:
 - *uly bolmadyk taslama*. Taslamany işläp düzüjileriň topary 10-dan köp bolmadyk adam sanyny öz içine alýar. Taslamanyň dowamlylygy 3-den 6 aýa çenli;
 - *ortaça taslama*. Taslamany işläp düzüjileriň topary 20-den 30 adama çenli öz içine alýar. Taslamanyň dowamlylygy 1-den 2 ýyla çenli;
 - *uly gerimli taslama*. Topar 100-den 300 adama çenli öz içine alýar. Taslamanyň dowamlylygy 3-den 5 ýyla çenli;
 - *örän uly taslamalar*. Topar 1000-den 3000 adama çenli öz içine alýar. Taslamanyň dowamlylygy 7-den 10 ýyla çenli.

3.2. Taslamalary işläp taýýarlamak prosesi we oňa gatnaşyjylar

Programmany işläp taýýarlamaklygyň iki sany ýakyn baglanyşykly düşüňjeleri – bu «taslama» we «taslamany düzüji topar» ýa-da ýöne «topar». Taslama toparlar bilen amala aşyrylýar, topar bolsa taslamalaryň üstünde işleýär.

Esasy toparlary indikilere: ulanyjylara, dolandyryjy komitete, maslahat beriji häsiýetli geňeşleriň birnäçesine we işläp düzüjileriň toparyna bölmek bolýar. Esasy toparlaryň arasyndaky maglumat akymlary 3.1-nji suratda getirilen we olary şeýle düşündirip bolar.

- dolandyryş komiteti toparlardan olaryň meýilnamalaryny alyp, olary tassyklaýar ýa-da tassyklamaýar;
- önümi işläp düzüji topar önümiň taslamasyny düzýän birnäçe komponentleriň kodlaryny birikdirýär. Bellibir wagtda esasy kodlaryndan, ulanyjylar üçin indiki görnüş (wersiýasy) şahalanyp bölünýär we oňa görnüşleriň topary jogap berýär;
- ulanyjylar (buýrujylar) öz talaplaryny we isleglerini teklipl edýärler we komponenti düzüjilerden önümiň prototipini, görnüşleriň toparyndan bolsa önümiň dürli görnüşlerini alýarlar.



3.1-nji surat. İşläp düzmeklik prosesine gatnaşyjylar

Aýratyn işleýän toparlara aýratynlykda seredip geçeliň:

- *ulanyjylar (buýrujylar)* işläp düzmek prosesine çekilen esasy gatnaşyjylar bolup durýarlar. Adatça, ulanyjy bolmasa, önüm hem bolmaýar. Ulanyjylaryň sanyny artdyrmakda gyzyklanma bildirýän iri kompaniýalar ulanyjylar bilen gatnaşyklary dürli ýollar bilen goldaýarlar;
- *dolandyryş komiteti* ösdürmegiň strategiýasyny we ösüş ugurlaryny işleýär, meýilnamalary tassyklaýar, önümiň indiki görnüşini öndürmek zerurlygyny kesgitleýär;
- *arhitektura boýunça geňeş* önümleriň parallel ösdürilmegi üçin maglumaty we tehniki bazisleri üpjün edýär, üýtgeşmeleriň arhitektura hemişelikligine gözegçilik edýär;
- *ýaşulularyň geňeşi* has tejribeli, hormatly işgärlerinden düzülýär we mümkin ýüze çykjak meseleleri tapmak hem-de gelejegi çaklamak üçin dolandyryş komiteti bilen çagyrylýar;
- *ulanyjylary goldaýan topar* müşderileriň çylşyrymly tehniki soraglaryna operatiw jogap berýär, şikaýatlary ýygnaýar we programma önümlerine «ýamalary» düzýär.

Programma önümini düzüjileriň toparynyň iki görnüşü bar:

- önümi işläp düzüjileriň topary – önümiň taslamasyna başlaýar, barlag geçirýär we ony ösdürýär. Toparyň baştutany bolup programma önüminiň taslamasynyň menejeri çykyş edýär. Bu topar önümi bilelikde düzmeklige gatnaşýan ähli funksional bölümlere gatnaşyjylardan düzülen;
 - düzüji injenerden (programmirlmegiň inženeriýasy boýunça hünärmenlerden we programma düzüjilerden);
 - tehniki ýazyjylardan;
 - testirleme inženerlerinden;
 - hil inženerlerinden;
 - önümi alyp barmak boýunça hünärmenlerinden;
 - önümi satmak boýunça hünärmenlerinden düzülen.

Topar önümi doly taslaýar we işläp düzýär. Adatça, işläp düzüjileriň topary komponentleri işläp düzüjileriň birnäçe toparlaryny hem öz içine alýar:

- komponenti işläp düzüji topar komponenti ýa-da häsiýeti döretmek we ösdürmek üçin niýetlenen. Topara, adatça,

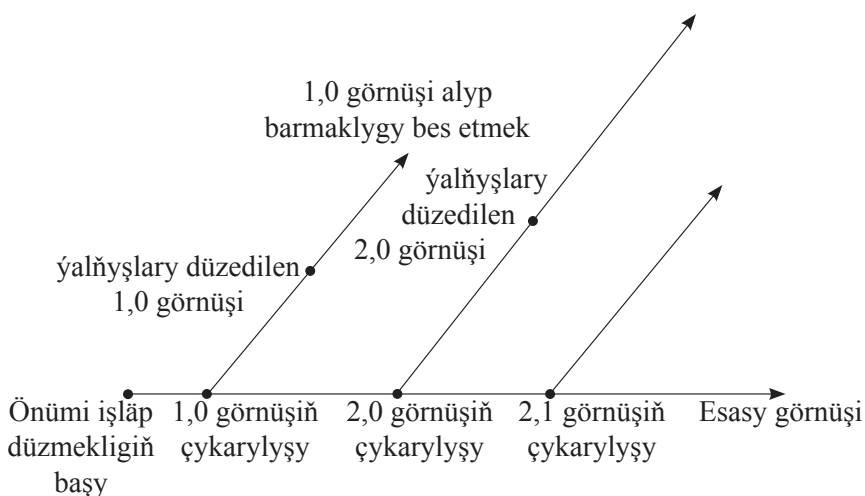
komponentiň taslamasynyň menejeri ýolbaşçylyk edýär. Oňa indikiler girýär:

- düzüji inžener;
- testirlemek inženeri;
- tehniki ýazyjylar.

Topar önümi taslamak tamamlanan badyna başlaýar. Hut şu topar önüm satylyp başlanda, ulanyjylar bilen tapylan ýalňyşlary düzedip, önümi alyp barmak bilen meşgullanýar.

Şeýle hem görnüşi taýýarlap çykarmaklyga gönükdirilen toparlar hem bar:

- *Görnüşleriň topary* önümiň alfa-görnüşlerini, betta-görnüşlerini we ulanyja ilkinji üpjünçiligi Internet serweriň sahypalarynda, disklerde ýa-da beýleki maglumat görterijilerde ýerleşdirmeklige we öndürmeklige jogap berýär. Görnüş bir ýa-da birnäçe önümi saklap biler. Görnüşleriň topary işläp düzüjiler toparynyň agzalaryndan düzülýär we programma önüminiň görnüşiniň menejeri ýolbaşçylyk edýär. Topar önümi işläp taýýarlamak we eltmek tertibini dolandyrýar, önümleri birleşdirmek we saýlanan görterijide paketleri ýygnamak barada çözümler kabul edilýär. Adatça, önüm we onuň görnüşü käbir simwoliki atlaryna eýe bolýar.



3.2-nji surat. Programma önüminiň görnüşleriniň yzygiderliligi

- *uýgunlaşdyrmak topary* belli dil we medeni gurşawy üçin önümiň uýgunlaşdyrmak (adaptasiýa) meselelerini çözüýär. Şunlukda, programmanyň ähli resminamalary we programmanyň ähli ýazgylary, maglumatlary we habarlary önümi uýgunlaşdyrmak diline geçirilýär.

Programma önüminiň görnüşleriniň yzygiderliligi 3.2-nji suratda görkezilen.

Programmanyň diňe bir esasy üýtgedilýän görnüşi bolýar we hut şondan ulanyjylar üçin niýetlenen beýleki görnüşler alynýar. Eger programmanyň täze görnüşi düýpli üýtgeşmeleri öz içine alýan bolsa (mysal üçin, täze interfeýs), onda onuň «esasy» nomeri bir birlik artýar (nokadyň öňündäki san). Eger üýtgeşmeler ujypsyz bolup, fukSIONALLYGYň käbir üýgeşmelerine jemlenýän bolsa, onda «ikinjilik» nomeri bir birlik artýar (nokatdan soňky san). Şahalanýan görnüşlere hem düzedişler girizilip bilner. Emma diňe esasy görnüşi ösdürilip, düýpli üýtgemeleri saklap biler, galan görnüşleri bolsa, alnyp barylýar.

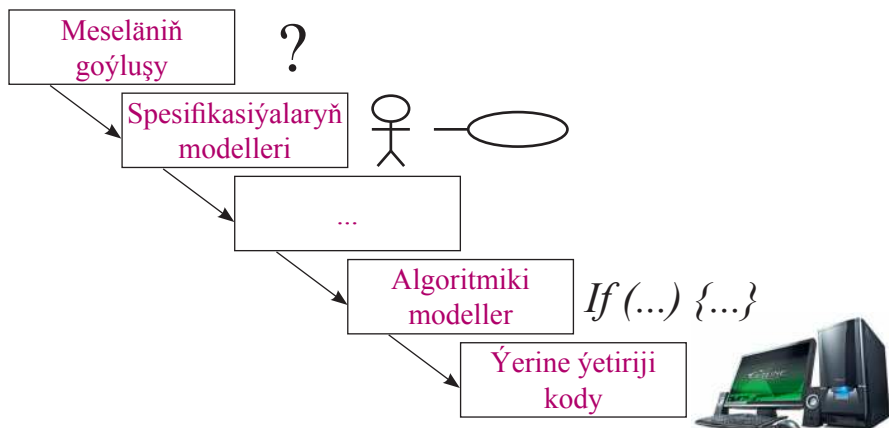


Profesional programmirlemegiň esasy meselesi

Profesional programmirlemegiň esasy meselesini ýokary hilli programma önümini döretmek hökmünde kesgittläliň. Bir söz bilen aýdanymyzda, onuň özeni bolup modelleriň yzygider transformasiýasynyň amala aşyrylyşy çykyş edýär. Meseläniň umumy goýulyşyndan we spesifikasiýalaryň has anyk modellerinden başlaýarys hem-de algoritmiki model we maşyn diliniň buýruklarynyň modeli bilen tamamlasarys (4.1-nji surat). Programmirlemegiň maksady owadan modelleri ýa-da algoritmeleriň kodlaryny almak däl-de, programmanyň işiniň netijelerini almak. Programmirlemegiň esasy meselesi programmirlemegiň usulyýeti we tehnologiýasy, programmirleme dilleri we ulgamlary, arhitektura we amallar platformasy ýaly esasy ugurlaryň kömegi bilen professional derejede çözülýär.

Programmirlemegiň esasy ugurlaryny ulanmaklyga berilýän maslahatlar:





4.1-nji surat. Meseläniň goýluşyndan ýerine ýetiriji koduna çenli programmirlemegiň ýoly

1. Ilki bilen bir umumy filosofiki çemeleşme bilen birleşdirilen, usullaryň we konsepsiýalaryň toplumyny öz içine alýan programmirlemegiň usulyýetini kesgitlemek gerek. Programmirlemegiň usulyýeti professional programmirlemegiň umumy medeniýetiniň bölegi bolmalydyr. Ol bellibir taslamanyň üstünde işlenip başlananda saýlanylýar (meseläniň spesifikasiýasyna baglylykda) we taslamanyň amala aşyrylyşynyň dowamynda üýtgemeli däl.

2. Soňra programma önümi döredilýän wagtynda ulanylýan prosesleriň jemini kesgitleýän tehnologiýa çemeleşmegi saýlamak gerek. Öňden kesgitlenen usulyýet tehnologiýa çemeleşmede ulanyljak usullaryň jemini öz içine alýar. Tehnologiýa çemeleşme taslamanyň üstünde işlenilende üýtgäp hem biler.

3. Usulyýet bilen tehnologiýa saýlanan tehnologiýa çemeleşmäniň her bir prosesi üçin zerur programmirleme dillerini we ulgamlaryny kesgitleýär.

4. Tehnologiýa prosesler käbir apparat we amallar platformalarynda amala aşyrylýarlar. Apparat we amallar platformalary gurallaryň (programmirleme ulgamlaryň) bolmaklygyny we spesifikasiýany düýpli kesgitlep bilerler. Işläp taýýarlamalaryň köpüsünde platformalara baglylykdan saklanmaly, emma taslamalaryň birnäçesi (düzgün bolşy ýaly, ulgamlaryň taslamalaryň) köp derejede olary gowy bilmeklige daýanýar.

II. PROGRAMMIRLEMEGIŇ USULÝÝETI



5. Programmirlmegiň usulyýetiniň esasy düşüňjeleri we olaryň klassifisirilenmegi

5.1. Programmirlmegiň usulyýetiniň esasy düşüňjeleri we kesgitlemeleri

Ýokarda biz programmirlmegiň usulyýetini programma önüminiň ýaşaýyş siklinde ulanylýan we bir umumy çemeleşme bilen birleşdirilen usullaryň jemi ýaly kesgitledik. Bu günki gün usulyýetiň görnüşleriniň, aýratyn hem doly (programma üpjünçiliginiň ýaşaýyş sikliniň ähli tapgyrlaryny hasaba alýan) usulyýetleriň, örän köpdügi ni bellemek gerek. Hut usulyýet programma üpjünçiligini düzmekde ulanyljak dilleri we ulgamlary kesgitleýär, köplenç, haýsy tehnologiýa çemeleşmäni ulanmalydygyny maslahat berýär. Her bir usulyýet bilen onuň üçin käbir häsiýetlendiriji atributlary baglamak bolar:

- *filosofiki çemeleşme* (ýa-da esasy düzgün). Ol kesgitlemek üçin ýönekeý we usulyýetiň effektiwlik esasy çüşmesini kesgitleýär;
- ylalaşylan, baglanyşykly *usullaryň* modelleriniň köplügi. Hut şu usullaryň üsti bilen berlen usulyýet amala aşyrylýar;
- usullary goldaýan we olary has takyk kesgitlemäge mümkinçilik berýän *konsepsiýalar* (düşüňjeler, pikirler).

Usulyýet programmirleme döwründe ulanylýan bolsa, ony köplenç, *programmirlmegiň paradigmasy* – bellibir programmirleme dili bilen bagly däl pikirlenmegiň we programmirlmegiň usuly diýip atlandyryýarlar.

Programmirlmegiň kesgitli usulyýetleri kesgitli programmirlme diller bilen goldanýarlar. Emma bu berlen usulyýete häsiýetli däl dilleri ulanyp bolmaýandygyny aňlatmaýar, ýöne kynçylyklar ýüze çykyp biler. Usulyýetleriň dil tarapynyň gysgaça sintaksiz we semantiki beýanyny getireliň. Şeýle hem bu usulyýetleri goldaýan dilleri hem görkezeliň, emma ilki bilen käbir kesgitlemeleri getireliň.

Sintaksis – tekstleri emele getirmek düzgüni. Sintaksisi beýan etmek üçin Bekus-Nauryň belgileriniň giňeldilen formal ulgamy ulanylýar. Onuň kömegi bilen simwollaryň yzygiderlikleriniň haýsysy programma kabul ediljekdigini kesgitlemek aňsat. Bu ulgamda bir sintaktik kategoriýalar beýlekileriň üsti bilen aňladylýarlar. Meta ululyklar burç ýaýlaryna alnan sözleri ýa-da sözleriň toparyny emele getirýärler. Meta ululygyň bahasy diýip diliň esasy simwollarynyň gutarnykly yzygiderligine düşünilýär. Hut şu yzygiderliklerden programmalar düzülýärler. Mysal üçin «::=» simwoly «ýaly kesgitlenýär» diýmekligi aňladýar, «|» simwoly «ýa-da» aňladýar. Käbir konstruksiýanyň islendik gezek gaýtalanmagyny (şol sanda 0 gezek) «{»we«}» ýaýlarynda ýerleşdirmek bolar. Kwadrat ýaýlaryna «[]» alnan simwollar hökmany däl diýip hasap edeliň. Bekus-Nauryň notasiýasyndaky sintaksis kesgitlemeleriň köpüsi rekursiwdir. Şeýle ýagdaýlarda sintaksis kategoriýalar beýlekileriň üsti bilen diňe gutarnykly ýagdaýda kesgitlenýärler.

Semantika – teksti onuň ugradylan tarap bilen düşünmek düzgüni. Sintaksizden tapawutlylykda, semantika adaty ýagdaýda söz bilen tebigy dilde beýan edilýär. Emma semantikanyň formal beýanyna birnäçe çemeleşmeler hem bar. Esasy we taryhda birinji bolan çemeleşme – bu amallar çemeleşmedir. Onda semantika käbir hasaplaýjynyň (kompýuteriň) adalgalarynda beýan edilýär. Hasaplaýjynyň işiniň dowamynda programmanyň üýtgeýänleri bilen emele getirilýän programma ýagdaýy üýtgeýär. Käbir başlangyç ýagdaý, şeýle hem programmanyň ýerine ýetirilişiniň netijesi bar.

Programmirlemegiň dürli usulyýetleri dürli häsiýetli meseleleri çözmekde dürli utuşlary berýärler. Käbir giňden ýaýran meseleleri görkezeliň:

- sanlaýyn programmirlemeklige esaslanan ylmy-tehniki hasaplamalar;
- maglumatlar bazalaryna birleşmekligi we hasabatlary generirlemek bilen bagly bolan ykdysady hasaplamalar;
- ýerine ýetirilişi meýilnamalaşdyrmaklygyň barlagyny talap edýän häzirkiki wagt ulgamlary;
- tekstleri işlemek, simwollaryň setirleriniň üstünde manipulýasiýa etmek;

– grafiki ulanyjylyk interfeýsleri.

Bu utuşy ilki bilen programma üpjünçiligini işläp taýýarlamak üçin sarp edilen umumy çykdajylar boýunça bahalandyryp bolar. Bu kontekstde çözülýän meseleleriň gerimlilikiniň programmirleme usulyýetini saýlamaklyga täsirini bellemek gerek. Ýene-de bir az ähmiýetli däl faktor – bu döredilen programma üpjünçiliginiň işiniň häzirki zaman kompýuterlerde netijeliligidir. Şeýle hem käbir usulyýetler bellibir meseleler üçin örän zyýanlydyr.

5.2. Programmirleme usulyýetleriň klassifikasiýasy

Usulyýetiň özeni boýunça klassifisirleme

Biziň usulyýete bolan çemeleşmämiz öz usullary bilen usulyýetiň käbir özeniniň barlygyndan ybarat. Bu özen käbir goşmaça aýratynlyklary bilen doldurylýar. Usulyýetleriň özenleri algoritmleri beýan etmek usuly bilen kesgitlenýär. Usulyýetiň esasy özenlerini sanap geçeliň:

- imperatiw programmirlemegiň usulyýeti;
- obýekte gönükdirilen programmirlemegiň usulyýeti;
- funksional programmirlemegiň usulyýeti;
- logiki programmirlemegiň usulyýeti;
- çäklendirmelerde programmirlemegiň usulyýeti.

Görnüşi ýaly, usulyýetler dolandyrmagyň ädimleýin gözegçiligini kesgitleýän nawigasion usulyýetler bilen diňe netijeli talaplary kesgitleýän spesifikasion usulyýetlere çenli aralykda ýerleşýärler.

Usulyýetiň topologiki spesifikasy boýunça klassifisirlenmesi

Usulyýetiň topologiki spesifikasy (topologiýa) usulyýetiň anyklanan özenini almak üçin usullary saýlap almak ýoly bilen kesgitlenýär. Özen-«düypleriniň» her biri, «gowy» ýa-da «gowy däl» bolup bilýän käbir topologiýa bilen kesgitlenýän «goşmaçany» alyp bilýär. «Gowy» ýa-da «gowy däl» diýmeklikligiň kriterisi hökmünde programma üpjünçiligini işlemeklige sarp edilýän umumy çykdajylar çykyş edip biler.

Çykdajylar köpsanly faktorlaryň jemi bilen, şol sanda maglumatlaryň abstraksiýalary, dolandyrylmagy we modullylyk bilen kesgitlenýärler. Mysal üçin, global maglumatlardan we şertsiz geçiş

operatoryndan gaýra durmaklyk (käbir aýratyn ýagdaýlardan başga), modullaryň örän baglylygy we olaryň gowşak birikdirilmegi gowy topologiýa getirýär.

Mysal getireliň. Eger imperativ usulyýetde düzümlýýin programmirlеме usullary goldanylýan bolsa, onda *düzümleýin programmirlеме usulyýeti* ady bilen örän belli bolan düzümlýýin imperativ programmirlемegiň usulyýetini alarys. XX asyryň 70-nji ýyllarynda testirmek üçin çykdajylary azaltmaklygyň zerurlygy düzümlýýin programmirlемäge geçmekligiň esasy sebäpleriniň biri boldy.

Obýekte gönükdirilen usulyýetiň şowlulygyny onuň maglumatlaryň abstrakt görnüşlerine esaslanýan, gowy topologiýasynyň kesgitländigini bellemek gerek.

Amala aşyrylyş spesifikasiýasynyň usulyýeti boýunça klassifisirlеме

Özen «düýpleriniň» her biri berlen usulyýetiň apparat goldawynyň guramagyny kesgitleýän «goşulmany» alyp biler. Häzirki wagtda has belli guramalar diýip ikisini: merkezleşdirilen we parallel guramalary belläp bolýar.

Usulyýetleriň özenleri ilki başda merkezleşdirilen arhitekturalar üçin niýetlenilip oýlanyldylar. Soňra bolsa, parallel apparat amala aşyrylyşlar döredilip, olara eýýäm bar bolan usulyýetleri uýgunlaşdyryp başladylar. Parallel usulyýetleriň mysallary getireliň:

- imperativ parallel programmirlемegiň usulyýeti. Bu usulyýeti, adaty, gysgaça – parallel programmirlеме usulyýeti diýip atlandyrylar;
- logiki parallel programmirlемegiň usulyýeti.

Garyşykly usulyýetler

Garyşykly usulyýetler birnäçe usulyýetleriň usullarynyň birleşmesini öz içine alýarlar. Köplenç, funksional we logiki programmirlемegiň usulyýetleri birleşýärler. Şeýle hem obýekte gönükdirilen we logiki programmirlеме usulyýetlerini birleşdirmek boýunça ylmy-barlag işleri hem bar (<http://www.math.spbu.ru/user/tseytin/papers.thml>). Barlaglaryň birnäçesi programmirlеме usulyýetleriniň unifikasiýasy meselelerine bagyşlanan [Hoare, Jifeng 1998].



Imperativ programmírmegiň usulyýeti

Imperativ programmírmegiň usulyýeti – ädimleýin görnüşde hasaplaýjynyň ýagdaýyny yzygider üýtgetmek düzgüni bilen häsiýetlenýän çemeleşme. Şunlukda, üýtgemeleri dolandyrmak doly kesgitlenen we doly derejede gözegçilik edilýär.

Imperativ programmírmeme – bu apparat taýdan ilkinji bolup goldanan programmírmeme usulyýeti. Ol uzak wagtyň dowamynda ýeke-täk apparat binägöri bolan Neýmanyň klassyky modeline gönükdirilen.

Usullar we konsepsiýalar

Ýagdaýlary üýtgetmek usuly ýagdaýy yzygider üýtgetmeklikden durýar. Bu usul algoritm konsepsiýasy bilen goldanýar.

Akymy dolandyrmak usuly dolandyrmaklyga ädimleýin gözegçilik etmeklikden durýar. Bu usul ýerine ýetirilish konsepsiýasy bilen goldanýar.

Sintaksis we semantika

Esasy sintaksiz düşünjesi bolup operator çykyş edýär. Birinji topar – atomar operatorlar. Olaryň hiç bir bölegi özbaşdak bolup durmaýar (mysal üçin, baha dakmak operatory, şertsiz geçiş operatory, prosedurany çagyryş operatory we ş.m.). Ikinji topar – beýleki operatorlary bir täze uly operatora birikdirýän düzümlleýin operatorlar (mysal üçin, düzüml operatory, saýlaw operatory, sikl operatory we ş.m.).

Programmírmegiň imperativ modelini goldaýan ýönekeý diliniň operatorlary üçin sintaksiz beýanlaryň bölegi şeýle bolup bilerdí:

$\langle operator \rangle ::= \langle ýönekeý operator \rangle \mid \langle düzümlleýin operator \rangle$
 $\langle ýönekeý operator \rangle ::= \langle baha dakmak operator \rangle \mid \langle çagyryş operator \rangle \mid \langle yza gaýdyş operator \rangle$
 $\langle düzümlleýin operator \rangle ::= \langle düzüml operator \rangle \mid \langle şahalanmak operator \rangle \mid \langle sikl operator \rangle$
 $\langle baha dakmak operator \rangle ::= \langle üýtgeýän ululyk \rangle := \langle aňlatma \rangle;$
 $\langle çagyryş operator \rangle ::= \langle podprogrammynyň ady \rangle ([\langle parametrleriň sanawy \rangle])$

$\langle yza\ gaýdyş\ operator \rangle ::= return\ [aňlatma];$
 $\langle düzüim\ operator \rangle ::= begin\ \langle operator \rangle\ \{; \langle operator \rangle\};$
 $\langle saýlaw\ operator \rangle ::= if\ \langle aňlatma \rangle\ then\ \langle operator \rangle\ | \ if\ \langle aňlatma \rangle\ then\ \langle operator \rangle\ else\ \langle operator \rangle$
 $\langle sıkl\ operator \rangle ::= while\ \langle aňlatma \rangle\ do\ \langle operator \rangle$

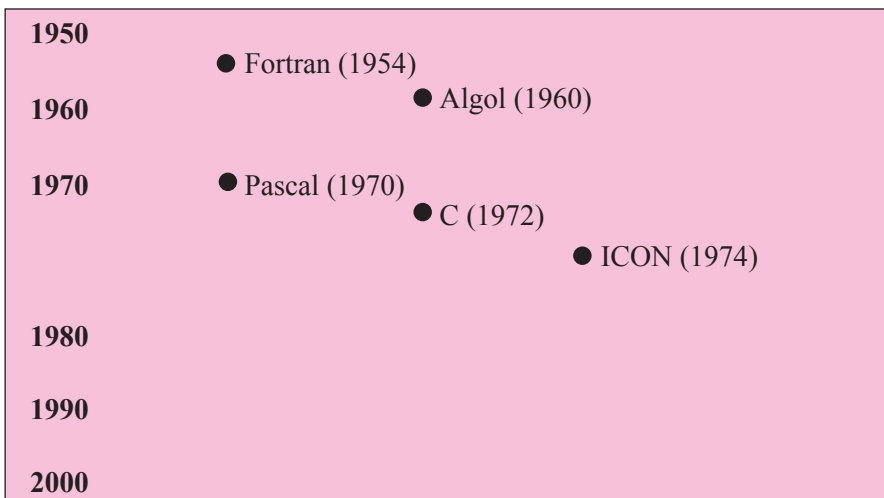
Imperativ programmirleme hasaplaýjynyň ýagdaýyny yzygider üýtgeýşiniň beýanyna esaslanan. Eger hasaplaýjy diýip häzirki zaman kompýuterlerine düşünsek, onda onuň ýagdaýy bolup ýadyň ähli öýjükleriniň bahalary, prosessoryň ýagdaýy (şol sanda buýruga görkeziji) we ähli birikdirilen gurluşlaryň ýagdaýy çykyş ederler. Ýeke-täk maglumatlar strukturasy – çyzykly tertipleşdirilen salgyly öýjükleriň yzygiderligi («salgy-bahasy» jübütleri).

Bu hasaplaýyş modelini goldaýan diller hasaplaýjynyň ýagdaýlarynyň arasyndaky geçiş funksiýalaryny beýan etmekligiň kompakt serişdesi bolup durýarlar.

Usulyýetiň gysgaça semantikasy. Operatorlar olary gurşap alýan düzümläýin operatorynda görkezilen tertibi boýunça ýerine ýetirilýärler. Eger bu düzüim operatory bolsa, onda oňa girýän operatorlar ýazylyş tertibi boýunça ýerine ýetirilerler. Saýlama operatoryna girýän operatorlar logiki şertiň bahasyna laýyklykda ýerine ýetirilip ýa-da ýerine ýetirilmän bilerler. Atomar operatorlaryň ýerine ýetirmegi hasaplaýjynyň ýagdaýynyň degişli birlik üýtgemegine jemlenýär. Proseduranyň ýerine ýetirilmekliginiň dowamynda beýleki çagyryşlaryň döremegine hem ýol berilýär.

Düzümläýin programmirlmegiň adaty serişdesi – bu podprogramma (prosedura ýa-da funksiýa). Podprogrammalaryň parametrleri we ýerli kesgitlemeleri bolýar we rekursiw çagyrylyp bilner. Funksiýalar öz işiniň netijesi hökmünde hasaplan bahasyny yzyna gaýtarýarlar.

Eger berlen usulyýetde käbir meseläni, onuň netijelerini indiki meseläniň çözgüdinde ulanmak üçin çözmeli bolsa, onda tipiki çemeleşme şeýle bolar. Ilki bilen birinji meseläni çözüýän algoritmi ýerine ýetirilýär. Onuň işiniň netijeleri indiki algoritme belli bolan ýadyň ýörite ýerinde ýatda saklanylýar we onuň bilen ulanylýar.



6.1-nji surat. Programmirlämegiň imperatiw dilleri

Imperatiw programmirlämegiň dilleri

Imperatiw programmirlämegiň dilleri yzygider instruksiýalaryny ulanyp we olary dürli görnüşli maglumatlara ulanyp, maglumatlary ädimleýin düzgüninde dolandyryrlar. Programmirlämegiň ilkinji algoritmiki dili bolup, 1945–46-njy ýyllarda Konrad Zuse tarapyndan işlenilip döredilen Plankalkuel dili hasap edilýär.

Has meşgul we giňden ýaýran programmirlämegiň imperatiw dilleri 6.1-nji suratda getirilen we olaryň köpüsi 50–70-nji ýyllar aralygynda döredilendir. Suratda görnüşi ýaly 80–90-njy ýyllar aralygy boş. Sebäbi bu period täze paradigmalaryň köpelmek periody we arassa imperatiw diller bu aralykda döredilmändir.

ICON imperatiw dilinde aşakda getiriljek programma tekst bilen işlemekligiň tipiki mysalyny görkezýär. Imperatiw dilleri eksperimental we ylmy-barlag goşundylary işläp taýýarlamak üçin ulanmak amatly.

Imperatiw dilinde programmanyň mysaly

```
# Programma buýruk setirinde parametrleri bilen berlen sözleri okaýar;
# we olary elipbiý boýunça tertipleşdirip, her biriniň näçe gezek gaýtalan-
# -ýandygyny görkezýär.
procedure main (argv)
```

```

T:=table (0)           # tablisany doretmek
every T[!argv] +=1      # buyruk setirinin her bir elementi tablisada
                        # acar yaly hasaplayjy hokmunde ulanylyar
L:=sort(T)              # acardan we bahasyndan sanawy doredip
                        # tablisany tertiplesdiryar
while pair:=pop(L) do   # sanawdan nobatdaky elementi alyp, onun
write(pair[1], «:», pair[2]) # acaryny we bahasyyny cap edyar
end

```

Imperativ programmirmek häzirki zaman aparat serişdelerini dolandyrmak programmalaryny düzmek üçin has amatly. Sebäbi häzirki zaman kompýuterleriniň ählisi diýen ýaly imperativ we usulyýet ýeterlik derejede effektiv kody döretmäge mümkinçilik berýär.

Meseleleriň çylşyrymlylygynyň ösmegi bilen imperativ programmalary okamak kyn bolýar. Imperativ programmleme usulyýetine esaslanan örän uly programmalary (mysal üçin, kompilýatorlaryň) programmirmek we ýola goýmak prosesi köp ýyllara çekip hem bilýär.



7. Obýekte gönükdirilen programmleme usulyýeti

Obýekte gönükdirilen programmleme usulyýeti – bu obýekt dekompozisiýasyny ulanýan çemeleşme. Onda ulgamyň statiki düzümi obýektleriň we olaryň arasyndaky aragatnaşyklaryň adalgalarynda beýan edilýär, ulgamyň özüni alyp barşy bolsa obýektleriň arasynda habarlaryň alyş-çalşygynyň adalgalarynda beýan edilýär.

Obýekt düşüňjesiniň döremegine modelirleme, maglumatlaryň görkezilişi, grafiki interfeýsleri we ulgamlaryň programmleme täsir etdiler.

Usullar we konsepsiýalar

Obýekte gönükdirilen dekompozisiýa usuly obýektleri we olaryň arasyndaky aragatnaşyklary bellemekden ybarat. Usul inkapsulirleme, dowam etdirme we polimorfizm konsepsiýalary bilen goldanýar.

Maglumatlaryň *abstrakt görnüşleri usuly* – bu inkapsulýasiýanyň esasynda duran usul. Ol abstragirmek konsepsiýasy bilen goldanýar.

Habarlary iberme usuly obýektleriň arasynda habarlaryň alyş-çalşygynyň adalgalarynda ulgamyň özüni alyp barmaklygynyň beýanyndan ybarat. Usul habarlar konsepsiýasy bilen goldanýar.

Sintaksis we semantika

Arassa obýekte gönükdirilen programmirmemegiň hasaplaýyş modeli açyk diňe bir amaly obýekte habary amaly ugratmak goldaýar. Habarlaryň özi hem obýekt bolup duran parametrlerine eýe bolup bilerler. Habaryň özi hem obýekt bolup durýar.

Obýekt habarlary işläp taýýarlaýjylaryň toplumyna (usullaryň toplumyna) eýe. Obýekt meýdanlara – diňe berlen obýekte degişli üýtgeýän ululyklara hem eýedir, olaryň bahalary bolup beýleki obýektlere salgylanmalar çykyş edýärler. Obýektiň meýdanlarynyň birinde obýekt başyna salgylanma saklanylýar we berlen obýekt bilen işlenilmeýän ähli maglumatlar oňa iberilýär. Habarlaryň işläp taýýarlanmagyny hem-de täzedan salgylanmagyny üpjün edýän gurluşlary aýratyn bir obýekte jemleýärler we oňa berlen obýektiň synpy diýilýär. Obýektiň özüne bolsa bellenen synpyň obýekti diýilýär.

Obýekte gönükdirilen dillerinde *synp* häsiýetleri, atributlary we özüni alyp barşy meňzeş bolan obýektleriň köplüginin funksionirlenmegini we düzümini (strukturasyny) beýan edýär. Obýekt käbir synpa degişli bolup, öz içki ýagdaýyna eýe bolýar. Usullar – bu aktiw edip bolýan funksional häsiýetler.

Obýekte gönükdirilen programmirmekde üç sany esasy häsiýetleri kesgitleýärler:

- *inkapsulirleme* – maglumatlary gizlemek we maglumatlaryň abstrakt görnüşlerine meňzeş funksiýalaryň we maglumatlaryň utgaşdyrylmagy. Şeýle hem inkapsulirmemegiň üýtgeýänleriň atlaryny we bloklarda proseduralary lokallaşdyrmak bilen umumylygy örän köp;
- *dowam etdirmе* – dörän obýektleriň ierarhiýasyny düzmek. Bu ierarhiýa degişli obýektleriň her birinde ähli dördediji obýektleriň maglumatlaryna we usullaryna elýeterlik mümkinçiligi bar;

– *polimorfizm* – amala şol bir ady dakmak. Soňra bu amal obýektleriň ierarhiýasy boýunça ýokarlygyna we aşaklygyna bölünýär, özi hem ierarhiýanyň her bir obýekti bu amaly oňa laýyk usul bilen ýerine ýetirýär. Şeýlelikde, şol bir at, ýerine ýetiriliş usuly dürli synplarda berilýän dürli amallara berilýär.

Her bir obýektde onuň degişli bolan synpyna salgylanmasy bolýar. Habar kabul edilende, obýekt berlen habary işlemek üçin synpa ýüzlenýär. Habary işlemek üçin synpyň usullary bolmasa, dowam etdirmekligiň ierarhiýasy boýunça ýokarlygyna iberilip bilner. Eger habarlar üçin wakalary işläp taýýarlaýjy dinamiki ýagdaýda saýlanýan bolsa, onda wakalary işläp taýýarlaýjylaryň amala aşyryan usullaryna wirtual diýilýär.

Aşakdaky mysalda «nokat» synpynyň beýanynyň mysaly käbir abstrakt obýekte-gönükdirilen dilinde getirilen.

Obýekte gönükdirilen dilinde programmanyň mysaly

```
Object :subclass Point :fields {x,y};
```

```
Point :answer isnew :args {init_x, init_y} :body {
```

```
    Integer : new x :value {init_x};
```

```
    Integer : new y :value {init_y}
```

```
};
```

```
Point :answer getx :arg {} :body { ^x};
```

```
Point :answer gety :arg {} :body { ^y};
```

```
Point :answer setx :arg {new_x} :body { x :set new_x};
```

```
Point :answer sety :arg {new_y} :body { x :set new_y};
```

```
Point :answer move :arg {delta_x, delta_y} :body {
```

```
    x :add delta_x
```

```
    y :add delta_y
```

```
};
```

```
// Eger biz Point synpyny ulanjak bolsak, mysal ucin onun obýektlerini
```

```
// emele getirip we olar bilen islejek bolsak, onda kod seyle bolar;
```

```
Point :new A :value{0,0};
```

```
A :move {+2,-2};
```

Synplar berlen usulyýetde strukturirlemegiň adaty serişdesi bolup çykyş edýärler. Synplar nusganyň haýsy meýdanlary we usullary

	Arassa	Gibrid	Gysgaldylan
1960	● Simula (1962)		
1970	● Smalltalk (1972) ● Beta (1975)		
1980			● C++ (1983) ●
	● Self (1986)		
1990	● Cecil (1992)		● Java (1995)
2000			● C# (2000)

7.1-nji surat. Programmirlemegiň obýekte gönükdirilen dilleri

daşyndan elýeterlidigini, aýratyn habarlary nähili işlemelidini we ş.m. kesgitleýärler. Arassa obýekte gönükdirilen dillerinde daşyndan diňe usullar elýeterlidir.

Obýekte gönükdirilen programmirlemegiň dilleri

Obýekte gönükdirilen programmirlemegiň dilleri synplara degişli we inkapsulýasiýa, dowam etdirmе we polimorfizm häsiýetlerine eýe bolan obýektleri kesgitlemäge mümkinçilik berýän konstruksiýalaryny saklaýar (7.1-nji surat). Şeýle dilleri üç topara bölüp bolýar:

- *arassa diller*. Olar obýekte gönükdirilen usulyýeti has klassiki görnüşde goldaýarlar. Şeýle diller uly kitaphanany, şeýle hem ýerine ýetirmek wagtyny goldaýan serişdeleriň toplumyny saklaýarlar;
- *gibrid diller*. Olar obýekte gönükdirilen konstruksiýalaryny programmirlemegiň meşgul imperatiw diline ornaşdyrmak netijesinde alnan;
- *gysgaldylan (arassalanan) diller*. Olar gibrid dillerinden obýekte gönükdirilen nukdaý nazaryndan has howply we gerek däl konstruksiýalaryny ýok etmek netijesinde döredilen.

Bu usulyýet obýektleriň arasynda gatnaşyklary modelirmek üçin kuwwatly serişde bolup durýar. Aýratyn hem ulanyjynyň gra-

fiki interfeýsleriniň dürli elementleriniň arasyndaky aragatnaşygy obýektlerde aňlatmak aňsat we amatly. Bu usulyýeti wagta berk çäk-lendirmeler goýlan hakyky wagt düzgünindäki ulgamlar üçin ulan-mak maslahat berilmeyär.



Funksional programmirlemegiň usulyýeti

Funksional programmirlemegiň usulyýeti – bu ýeke-täk amaly – funksiýa, programmany böleklerе bölmek usulydyr. Fuksiýa üçin ady we bu at üçin funksiýanyň bahasyny hasaplaýan aňlatmany gi-rizmek, ýeke-täk kompozisiýa düzgünidir. Ol funksiýanyň superpo-zisiýa usuly ýaly şertleri kanagatlandyrýan programmalary düzmek usulydyr.

Usullar we konsepsiýalar

Applikatiwlik usuly boýunça programma- bu argumentlere funk-siýalary ulanmak amallaryndan düzülen aňlatma. Programma beýleki funksiýalary çagyrmaklygy aňladýan we bir biriniň içinde ýerleşen funksiýalaryň toparyndan düzülen. Usul *funksiýa* konsepsiýasy bilen goldanýar.

Rekursiw alyp barmak usuly funksiýa öz özüne gaýdyp gelmegi, öz-özünü gaýtalamaklygy aňladýar. Usul rekursiýa konsepsiýasy bi-len goldanýar.

Sazlanmaklyk usuly nusga boýunça täze programma obýektle-rini degişli aňlatmalaryň bahalary hökmünde aňsat döredip bolýan-dygyny aňladýar. Muňa diňe programma däl-de, islendik programma önümi aňlatma bolup çykyş edýändigini ýardam edýär.

Sintaksis we semantika

Funksional programmirleme imperatiw çemeleşmäniň alternati-walarynyň birini emele getirýär. Funksional programmirlemede wagt düşünjesi ýok. Programmalar aňlatma bolup çykyş edýärler, programmalaryň ýerine ýetirilişi bolsa, bu aňlatmalaryň hasaplanyl-magyndan ybarat.

Funksional programmirleme beýan edilende, düzgün bolşy ýaly, «giňeldilen lýambda-hasaplama» diýip atlandyrylýana seredilýär.

Onuň grammatikasy şeýle bolup biler:

```

<aňlatma> ::= <ýönekeý aňlatma> | <düzüm aňlatma>
<ýönekeý aňlatma> ::= <konstanta> | <ady>
<düzüm aňlatma> ::= <lýambda-abstraksiýa> | <ulanyş> | <kwalifisirlenen
aňlatma> | <şahalanma>
<lýambda-abstraksiýa> ::= lambda <ady> → <aňlatma> end
<ulanyş> ::= (<aňlatma> <aňlatma>)
<kwalifisirlenen aňlatma> ::= let <ady>=<aňlatma> {;
<ady>=<aňlatma>} in <aňlatma> end
<şahalanma> ::= if <aňlatma> then <aňlatma> end | if <aňlatma> then
<aňlatma> else <aňlatma> end

```

Giňeldilen lýamda-hasaplamada konstantalar bolup sanlar, kortezlar, sanawlar, öňden kesgitlenen funksiýalaryň atlary we ş.m. çykyş edip bilerler. Argumentlere öňden kesgitlenen funksiýany ulanmaklygyň hasaplanyşynyň netije bolup öňden kesgitlenen funksiýanyň bu «nokatdaky» bahasy bolar. Argumente lýamda-abstraksiýany ulanmaklygyň netijesi bolup argumenti lýamda-abstraksiýanyň «esasy bölegi» aňlatma goýmaklyk çykyş eder. Lýamda-abstraksiýalaryň özleri hem aňlatmalar bolup durýarlar, diýmek argumentler bolup bilerler.

Lýamda-abstraksiýalaryň diňe bir argumenti bolýar, ýöne şol bir wagtda funksiýalaryň, olaryň adaty görnüşinde, bir ýerli bolmaklygy hökmany däl. Birnäçe argumentli funksiýalary iki usul bilen almak bolar:

- birinji usulda argument bolup, kortez çykyş eden ýagdaýyny hasap edeliň. Mysal üçin,
 $apply = \lambda (f, x) \rightarrow (f, x)$ end
 ýazga şeýle düşünmeli
 $apply = \lambda y \rightarrow ((first\ y)\ (second\ y))$ end.
- ikinji usul hasaplanýan funksiýalaryň köplügi $X * Y \rightarrow Z$ iki taraplaýyn $X \rightarrow (Y \rightarrow Z)$ hasaplanýan funksiýalaryň köplüğine geçirilýär. Şeýle
 $apply = \lambda f \rightarrow \lambda x \rightarrow (f, x)$ end end.

Programmalaryň görnüşiniň korrektligini üpjün etmek üçin funksional dillerine ýola goýmaklygy goldamaklyga gönükdiri-

len görnüşleriň ýörite ulgamlaryny girizýärler. Düzgün bolşy ýaly, funksional dilleriň translyatorlary aňlatmalaryň görnüşini hiç hili görnüşler yglan edilmese-de, özbaşdak kesgitläp bilýärler. Şeýle indiki funksiýanyň

$add = \lambda x \rightarrow \lambda y \rightarrow x + y \text{ end end.}$

görnüşü $\text{number} \rightarrow \text{number} \rightarrow \text{number}$ bolar.

Funksional dilinde programmanyň mysaly

let

map=fun

$(f, []) \rightarrow [];$

$(f, [head|tail]) \rightarrow [f(head)|map(f,tail)]$

end fun;

square=fun

$x \rightarrow multiply(x,x)$

end fun

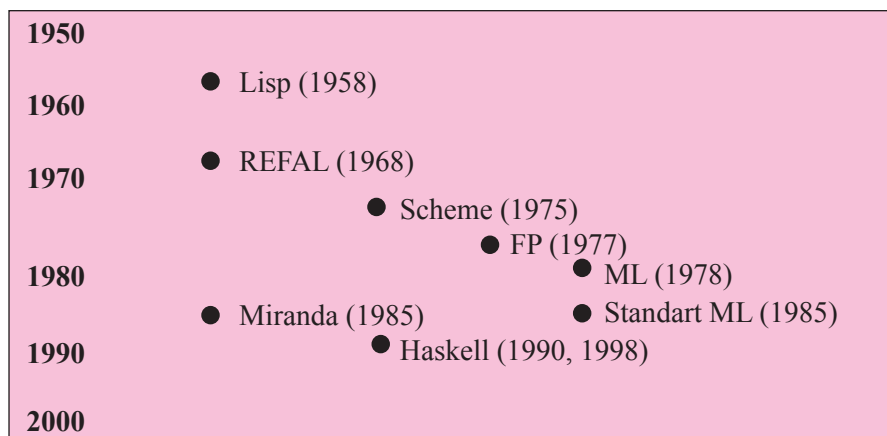
in

map(square, [1,2,3])

end let

Funksional programmirlenmegiň dilleri

Funksional dilleriň esasy spesifikasiýasy – bu funksiýalar öz arasynda maglumatlar bilen gös-göni alyş-çalyş edýärler, diýmek, hiç hili aralyk üýtgeýänleri-de, baha dakmaklygy-da ulanmaýarlar (8.1-nji surat).



8.1-nji surat. Programmirlenmegiň funksional dilleri

Üýtgeýänler bir gezek bahany alyp, ony hiç wagt üýtgetmeýärler. Bahany dakmak (mysal üçin, imperatiw usulyýetde) – bu hasaplaýyş prosesine wagty girizýän konstruksiýa. Baha dakmaklygyň sag tarapy hasaplanan bolmaly we diňe şondan soň onuň bahasy çep tarapy bilen baglaşdyrylýar. Netijede, funksional dillerinde sikller rekursiw funksiýalaryň apparaty bilen çalşyrylýarlar. Baha dakmaklygyň ýoklugy sebäpli, funksional diliniň programmalary hiç hili zyýanly netijeleri saklamaýarlar (8.1-nji suratda esasy funksional dilleri getirilýär).

Funksional programmirlеме, adaty, yzygider amallarynyň adalgalarynda kesgitlemegi kyn bolan meseleleri çözmek üçin ulanylýar. Bu kategoriýa emeli intellekt bilen bagly bolan meseleleriň ählisi diýen ýaly girýärler. Bu meselelere tebigy dili işlemek, ekspert maslahat beriji ulgamlar, göz bilen kabul etmek meseleleri we beýlekiler degişlidir.



9. Logiki programmirlемegiň usulyýeti

Logiki programmirlемegiň usulyýeti – bu çemeleşmede programma meseläniň beýanyňy faktlaryň adalgalary we logiki formulalary görnüşinde saklaýar, meseläniň çözüwini bolsa ulgam, logiki çykyş mehanizmleriniň kömegi bilen amala aşyryr.

Usullar we konsepsiýalar

Birmeňzeşlik usuly logiki subut etme mehanizmini ähli programma birmeňzeş ulanmaklykdan ybarat.

Unifisirleme usuly – maglumatlaryň strukturalarynyň dekompozisiýasy we dördilişi üçin nusga bilen deňeşdirmek mehanizmi.

Sintaksis we semantika

Logikada teoriýalar aksiomalaryň we çykaryş düzgünleriniň kömegi bilen berilýärler. Prolog logiki programmirlемegiň bazis dilinde hem edil şuna meňzeş. Onda aksiomalar faktlar diýip atlandyrylar, çykalyş düzgünleri bolsa forma boýunça «Hornýň dizýunktlary» diýip atlandyrylýan $A \leq B_1 \& \dots \& B_n$ görnüşli tassyklamalara çenli gysgaldylan. Prolog dilinde şeýle tassyklamalary indiki görnüşde ýazmak kabul edilen:

a: -b1,...,bn.



Faktlar (aksiomalar) boş «iberişli»: a düzgün hökmünde peýda bolýarlar.

Tassyklamalarda üýtgeýänleri baş harp bilen başlanýan identifikatorlar bilen bellemek kabul edilen. Prolog dilinde ýönekeý programmanyň mysaly şeýle ýazylyp bilner:

member (X, [X|_]).

member (X, [_|T]) : – member(X,T).

Bu programmany şeýle okap bolar. Eger-de ol sanawyň başy bilen gabat gelýän bolsa ýa-da sanawyň soňunyň agzasy bolsa X sanawyň agzasy bolýar. Bu programmada diňe bir predikat – member yglan edilen.

Adatça, Prolog-ulgamy ulanyjy bilen sorag-jogap düzgüninde işleýär. Subut edilmeli tassyklamany klawiaturadan girizmeli. Prolog translýatorlaryň kompilirleýän görnüşlerinde tassyklamalary girizmek üçin ýörite sintaksis serişdeleri bar. Bu tassyklamalary Prologda maksatlar diýip atlandyrmak kabul edilen.

Ulgama ýönekeý sorag bereliň: «2 san [1,2,3] sanawyň agzasy bolýarmy?»

Onuň üçin indikini girizeliň:

? – member(2, [1,2,3]).

Prolog-ulgamy ilki bilen member predikaty üçin birinji düzgüni ulanmaklyga synanyşýar, 2 sany [1,2,3] sanawyň başy bilen deňeşdirýär. Bu deňeşdirme üstünlikli amala aşyrylmaýar. Ulgam ikinji düzgün boýunça, 2 we [2,3] argumentli member predikatyny rekursiw çykarmaklygy dowam etdirer. Bu rekursiw çagyrylyşda birinji düzgüni ulanyp bolar (sebäbi 2, [2,3] sanawyň başy bilen gabat gelýär), onda Prolog-ulgamy şeýle jogaby çykarar:

yes →

Ýa-da Prolog-ulgamyna şeýle soragy hem berip bolar, mysal üçin, haýsy X üçin, onuň [1,2,3] sanawynyň agzasy bolýandygy dogry?

? – member(X, [1,2,3]).

Ulgam yzygider şeýle jogaplary berer:

X=1 →

X=2 →

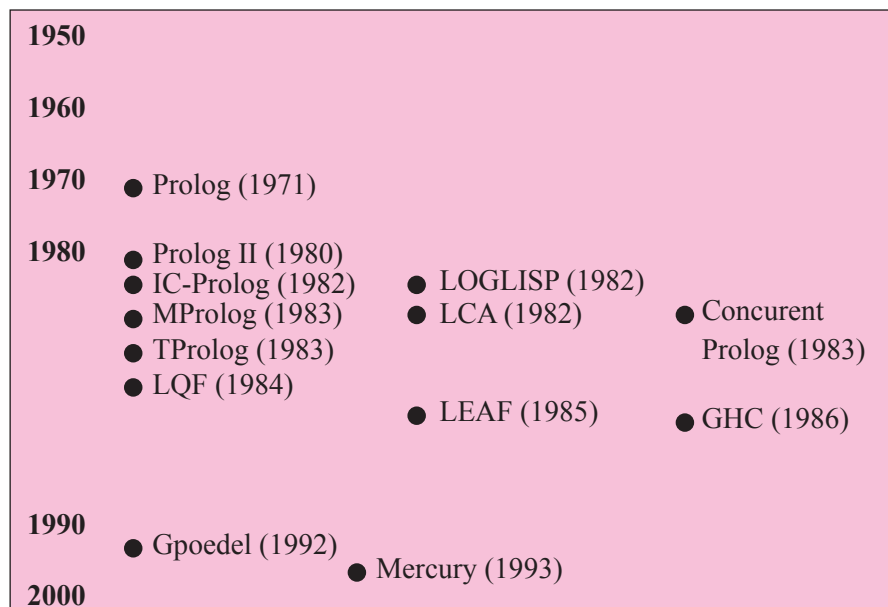
X=3 →

no.

Prolog-ulgamy tassyklamany subut etmek üçin unifikasiýa we rezolýusiýa usullaryny ulanýar. *Unifikasiýa* – bu üýtgeýänleri saklaýan iki sany erkin termleriň deňeşdirmesi. Bu deňeşdirme iki sany meňzeş termleri almak üçin bu üýtgeýänlere baha dakyp bolýanlygyny anyklamak maksady bilen amala aşyrylýar. Mysal üçin, $f(X,2)$ we $f(1,Y)$ termleriň unifikasiýa. Bu ýerde X,Y – üýtgeýänler şeýle jogaby berer $X=1,Y=2$. $f(X)$ we X termleriň unifikasiýa şowsuz amala aşyrylar. Rezolýusiýalar usuly Hornyň dizýunkt ugratmasyna girýän aýratyn tassyklamalaryň yzygider subut edilmeginden ybarat. Diýmek, rezolýusiýalar usulyny $a :-b, c$ kadasyna we a tassyknamasy-na peýdalanmak, b we c tassyklamalaryň yzygider subudyna getirer.

Logiki programmirlemegiň dilleri

Logiki programmirlemegiň dilleri meseläniň beýanyňy faktlaryň adalgalarynda we logiki formulalarynda ýerine ýetirmeklige mümkinçilik berýän konstruksiýalaryny saklaýarlar, meseläniň çözgüdi ni bolsa, ulgamyň özi logiki çykaryş mehanizmleriň kömegi bilen amala aşyrylar (9.1-nji surat). Prolog dili logiki programmirlemegiň dilleriniň düýptutany hasaplanýar we onda üç ugry belläp bolar:



9.1-nji surat. Programmirlemegiň logiki dilleri

- diliň modifikasiýalary (has kuwwatly logiki serişdeleri ulanmak);
- funksional ugur (funksional diller bilen kombinasiýalar);
- parallel ugur (logiki programmirlleme öz esasy boýunça parallel).

Logiki programmirllemegiň meseleleriniň klasy funksional programmirllemegiň meseleleri bilen gabat gelýär.



10. Çäklendirmelerde programmirllemegiň usulyýeti

Çäklendirmelerde programmirllemegiň usuly. Bu çemeleşmede programmada çözüwiň görnüşi we gözlenýän çözüwiň bahasyna çäklendirilmeler önünden kesgitlenýär. Çözüw ulgam bilen tapylýar. Usulyýet programma önümini we çäklendirme komponentini integirlenýän iki derejeli arhitekturany hödürleýär. Çäklendirmeleriň komponenti esasy amallary üpjün edýär we çäklendirme ulgamynyň fundamental häsiýetlerindäki netijeler ulgamyndan ybarat. Çäklendirme komponentiniň daşyndaky amallar dil-programma komponenti bilen amala aşyrylýarlar.

Usullar we konsepsiýalar

Hasaplamalary beýan etmekligiň usulynda programmirlleme dilinde ýazylan programma düşünjeleriň we meseleleriň beýanyny saklaýar. Usul *model* konsepsiýasy bilen goldanýar.

Sintaksiz we semantika

Çäklendirmelerde programmirlleme – bu meseläniň berlişini adalgalarda programmirllemek.

Meseläniň goýluşy – bu üýtgeýänleriň gutarnykly toplumy $V = \{v(1), \dots, v[n]\}$, olara degişli bahalaryň gutarnykly (sanalýan) köplügi $D = \{d(1), \dots, d[n]\}$, we çäklendirmeleriň toplumy $C = \{c(1), \dots, c[n]\}$. Çäklendirmeler tassyklamalar görnüşinde getirilen. Olaryň parametrleri hökmünde V toplumyň käbir köplüğinden $v[j]$, $j=1 \dots m$ üýtgeýänler çykyş edýärler. Şeýle meseläniň çözüwi – ähli $c[j]$ çäklendirmeleri kanagatlandyryan üýtgeýänleriň bahalarynyň toplumy.

Programmanyň semantiki ýerine ýetirilişine üýtgeýänleriň bahalarynyň tapylyşy hökmünde seredilýär. Şunlukda, aýratyn çäklendirmeleri kanagatlandyrmaklygyň tertibiniň tapawudy ýok we ulgam bu ýagdaýda, adaty, şowsuz ýerine ýetiriliş ýagdaýynda yza gaýtmaklygy minimizirmek maksady bilen bahalary tapmak yzygiderligi optimizlemeklige ymtylýar.

Çäklendirmelerde meseläni minimal zerur bolan çäklendirmeleriň toplumy baradaky mesele hökmünde seretmek hem bolýar. Bu käbir ýagdaýlarda tükeniksiz köp çözüwleri anyk görnüşde beýan etmeklige mümkinçilik berýär. Şeýle $X:\text{integer}, X>2, X<4$ mesele üçin minimal zerur bolan çäklendirmeleriň toplumy bolup $X=3$ bolýar. $X,Y:\text{integer}, X*2=Y$ mesele üçin şeýle toplum bolup $X:\text{integer}, Y \bmod 2=0$ çykyş eder.

Simwol hasaplama ulgamlary käwagt özleri hem çäklendirme bolup duran «kadadan çykmalary» ulanmaga mümkinçilik berýärler. Indiki ýönekeý talaba

Assume $X>0$.

when $X+1<10$?

ulgamlar şeýle jogap berip bilerler:

X in $(1..8)$.

Düzgün bolşy ýaly, şeýle ulgamlar trewial däl matematiki tassyklamalary subut edip bilýärler.

Çäklendirmelerde programmirlemegiň dilleri

Çäklendirmelerde programmirlemegiň dilleri programmada çözüwiň görnüşini we gözlenýän çözüwiň bahasyna çäklendirilmeleri kesgitlemäge mümkinçilik berýärler. Çözüwi ön belläp geçişimiz ýaly, ulgam bilen tapylýarlar. Bu görnüşli diller geçen asyryň 80-nji ýyllarynda has meşgul boldular (10.1-nji surat).

УТОПИСТ (Uniwersal Tekst Beýanlarynyň Terminleri) programmirleme dili düşüňjeleri we meseleleri beýan etmek üçin niýetlenen. Bu dil ППИЗ gurallar programmirleme ulgamy üçin eslandyryjy. Diliň proseduralaýyn bölegi hem bar, emma onuň esasy täsirliги beýanlaryň hasabyna ýetilýär. Aşakda Omuň kanunynyň esasynda käbir bahalarynyň hasaplanylşynyň УТОПИСТ dilinde meseläniň modeli we goýluşy getirilýär.

1950			
1960	Sketchpad (1963)		
1970			
1980	UTOPIST (1980)	Thinglab (1980)	IDEAL (1981)
	OPSS (1987)	Bertrand (1988)	
1990			
2000	OPL (1998)		

10.1-nji surat. Çäklendirmelerde programmirlemegiň dilleri

Çäklendirmelerde programmirleme dilinde programma

ПУСТЬ ‘

CXEMA: (I1,I2,I3,U1,U2,U3,R1,R2,R3: БЕЩ ‘;

УРАВ’ $U1=I1 \cdot R1$

УРАВ’ $U2=I2 \cdot R2$

УРАВ’ $U3=I3 \cdot R3$

УРАВ’ $I1+I2+I3=0$);

CXEMA1: CXEMA

$R1=16, R2=32, R3=5,$

$U1=50, U2=-28;$

ДЕЙСТВИЯ ‘

НА’CXEMA1 ВЫЧИСЛИТЬ’U3;

Çäklendirmelerde programmirleme meselelerine amallary barlamak we emeli intellekt meseleleri degişli. Şeýle meselelerde, köplenç, çözgütleriň käbir giňişligi ulanylýar we onuň daralmagy bilen gerekli netije alynýar. Şeýle daralmalary, adatça, çäklendirmeler diýip hasaplap bolar.



11. Düzümleýin imperatiw programmirlemegiň usulyýeti

Düzümleýin imperatiw programmirlemegiň usulyýeti – bu çemeleşmede programma üpjünçiligini işläp taýýarlamak üçin sarp edilýän umumy çykdajylaryň azaltmagyna gönükdirilen imperatiw programmalaryň topologiýasyny bermeklik göz önünde tutulýar.

Taslamalaryň modelleriniň we programma kodlarynyň gowy strukturirlenmegi köp ýalňyşlyklardan gaçmaklyga we çykdaýlary azaltmaklyga mümkinçilik berýär. Bu usulyýetde global maglumatlardan, köp ýagdaýlarda, şertsiz geçiş operatoryndan gaçmaklyk, güýçli baglanyşylyk häsiýetli modullary işläp düzmek we olaryň beýleki modullardan garaşsyz işlemekligini üpjün etmek gowy topologiýany emele getirýärler. Bu çemeleşme gurulmagyň iki düzgünine esaslanýar:

- meseläni ýokardan aşaklyga çözmek algoritminiň yzygider kompozisiýasy;
- düzümläýin kodlamaklygy ulanmak.

Bu usulyýet imperatiw usulyýetiň möhüm ösüşidir.

Düzümläýin çemeleşmäniň döredijisi bolup, Edsger Deýkstra hasap edilýär. Düzümläýin programmirlenmegi programmalary subut etmek usullary bilen birikdirmek (häzirki wagtda köpçülik programmirlenmek üçin düýbünden ulanmak bolmaýar) synanyşygy hem Deýkstra degişlidir.

Usullar we konsepsiýalar

Ýokardan aşaklygyna algoritmiki dekompozisiýa usuly boýunça meseläniň goýluşy, has umumy meseleden başlap, ädimläýin detali-zasiýa edilýär. Bu usul gowy strukturirlenmegi üpjün edýär we *algoritm* konsepsiýasy bilen goldanýar.

Programmanyň böleklerini modullaýyn gurnamak usuly boýunça programma modullar diýip atlandyrylýan ýörite komponentlere bölünýär. Usul modul konsepsiýasy bilen goldanylýar.

Düzümläýin kodlamak usuly boýunça kodlamakda üç esasy dolandyryjy konstruksiýalar ulanylýar. Belgiler we şertsiz geçiş operatory kyn yzarlanýan aragatnaşyklar bolup çykyş edýärler. Şonuň üçin olary mümkin bolsa, ulanmak maslahat berilmeýär. Usul *dolandyryş* konsepsiýasy bilen goldanýar.

Düzümläýin programmirlenmegiň dilleri

Imperatiw programmirlenmegiň klassiki usulyýetinden tapawudy programmalaryň düzümläýin görnüşinde we operatorlaryň kompozisiýasynyň üç usulynda: iki ýa-da köp operatorlaryň yzygiderligi, dihotomiki saýlawy we gaýtalanmagydyr.

Eger-de while B do S end gaýtalama konstruksiýasyny (sikl operator) alsak, onda ol wajyp häsiýete – operator ýerine ýetirilenden soň B şerti S operatora garamazdan «ýalan» baha eýe bolar. Şeýle konstruksiýalar ygtybarly programmalary düzmäge mümkinçilik berýärler we testirmek bilen ýola goýmaklyga sarp edilýän wagty tygşytlýarlar.

Şeýle hem ýene-de bir ähmiýeti az bolmadyk aýratynlyk bu şertsiz geçiş operatoryny ulanmazlykdyr.

Imperativ usulyýetini goldaýan programmirleme dilleriň ählisinde diýen ýaly programmany bu usulyýet boýunça düzmek hem bolar. Dürli programmirleme diller bilen goldanýan şertsiz geçiş operatory barada dört sany esasy pikirler 11.1-nji tablisada getirilýär.

11.1-nji tablica

Goto operator barada dört sany esasy pikirler

Nukdaýnazar	Goto çalşyryjy	Goto ulanmak	Dilleriň mysallary
Ortodoksal (Deýkstra)	Ýok	Ýok	BLISS, PDL
Diňe goto çalşyryjylar	Hawa	Ýok	Java, Modula-2
Deýkstranyň tankytçylary	Ýok	Hawa	Basic
Hem goto çalşyryjylar hem goto	Hawa	Hawa	C, Ada

Bu usulyýet boýunça çözülýän meseleleriň topary imperativ usulyýet üçin meseleleriň topary bilen gabat gelýär. Bu usulyýet boýunça programmalary çylşyrymly düzmek aňsat bolýar, sebäbi olary kabul etmek we seljermek aňsat bolýar.



12. Imperativ parallel programmirlmegiň usulyýeti

Imperativ parallel programmirlmegiň usulyýeti – bu çemeleşmede programmalaryň bellenen fragmentlerini parallel amala aşyrmak üçin aýyk konstruksiýalaryň ulanylmagy hödürlenýär.

Usullar we konsepsiýalar

Ýerine ýetirilýän koduň *sinhronizasiýasy* usuly boýunça bir wagtda ýerine ýetirilýän koduň bölekleriniň aragatnaşygyny amala aşyrmak üçin ýörite atomiki amallary ulanýarlar. Usul *sinhronizasiýanyň ýönekeýligi* konsepsiýasy bilen goldanýar.

Sintaksis we semantika

Parallel programmirlеме usulyýetinde operatoryň göni analogy bolup proses çykyş edýär. Onuň imperatiw usulyýetinden esasy aýratynlygy – prosesleriň parallel ýerine ýetirilip bilijiligidir.

Parallel usulyýet onuň amala aşyrylýan platformasyna bagly. Parallelizm merkezleşdirilen arhitekturada hem, paýlaşdyrylan arhitekturada hem apparatlaýyn amala aşyrylyp bilner.

Umuman parallelizm goşundylaryň, prosesleriň, programmalaryň, podprogrammalaryň, siklleriň we operatorlaryň şol bir wagtda amala aşyrylmagyndan ybarat. Bölmekligiň derejesine baglylykda parallelizmiň indiki derejelerini bölmek bolar:

- mikro buýruklaryň derejesinde parallelizm;
- operatorlaryň derejesinde parallelizm;
- siklleriň we iterasiýalaryň derejesinde parallelizm;
- podprogrammalaryň derejesinde parallelizm;
- dolandyryş akymalaryň derejesinde parallelizm;
- prosesleriň derejesinde parallelizm;
- goşundylaryň derejesinde parallelizm.

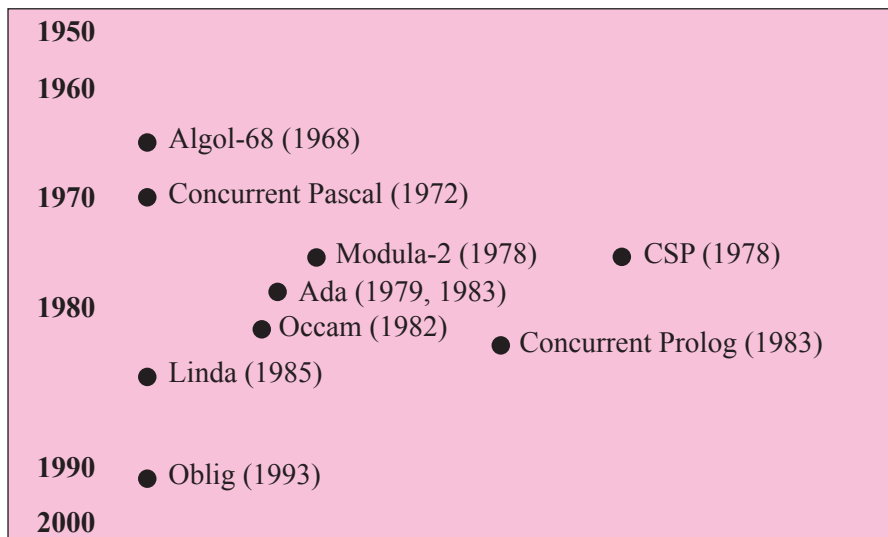
Bu usulyýetde ýokary derejeli strukturirlemekligiň adaty serişdesi – proses. Prosesler parallel işläp bilerler.

Parallel prosesleriň semantiki özara gatnaşygy kanallar bilen birleşdirilen gurluşlaryň torunyň işine meňzeş. Şeýle adalgalarda mysal üçin, sanlaryň örän uly sanyny çalt goşmaklyk üçin parallel prosesleriň ulgamyny taslap, beýan edilen sintaksisde bu ulgamy örän aňsat beýan edip bolar. Her bir hasaplaýjy onuň hasaplaýyş modeli üçin tipiki amallary ýerine ýetirýär (mysal üçin, imperatiw hasaplaýjy bir ýagdaýdan beýleki ýagdaýa geçip durýar). Haçanda proses «akymdan bahany almak» instruksiýa duşanda, ol akym boş wagty garaşmak ýagdaýyna geçýär. Akymda baha emele gelen wagtynda, proses ony okaýar we işi dowam edýär.

Parallel programmirlemegiň dilleri

Parallel programmirlemegiň dilleri programmalaryň belle-nen böleklerini parallel ýerine ýetirmek üçin anyk konstruksiýalary ulanýarlar. Parallel hasaplaýyş ulgamlary üçin programmirlemeklige birnäçe dil çemeleşmeleri bar (*12.1-nji surat*):

- programmirlemegiň parallel dilinde programmirlemek. Bu ýagdaýda diller şeýle bolup bilerler:
 - uniwersal (mysal üçin Ada);
 - parallel dilinde programmalary netijeli translirlemäge mümkinçilik berýän kompýuterleriň bellibir görnüşleri üçin (mysal üçin, Occam ilkibaşda transkompýuterler üçin işlenildi);
- diliň parallel konstruksiýalary bilen giňeldilen we giňden ýaýran programmirleme dilinde programmirlemek (mysal üçin, C, C++, Pascal);
- kompilyatora programmalaryň dilinde goşmaça görkezmeleri ulanmak bilen programmirleme (mysal üçin, OpenMP standarty boýunça);
- kommunikasion kitaphanalary we prosessor aragatnaşygy guramak üçin interfeýsleri ulanmak bilen giňden ýaýran programmirleme dilinde programmirlemek. Bu ýagdaýda paralel-



12.1-nji surat. Programmirlemegiň parallel dilleri

lizm konstruksiyalary dil derejesinden operasion ulgamynyň derejesine çykarylan;

- kompilyatorlar ýaly gurallar bilen yzygider programmalary awtomatiki paraleleşdirmekligiň serişdelerini ulanmak.

(12.1-nji suratda parallel ýerine ýetirilişiň anyk funksiýalaryny saklaýan giňden ýaýran programmirleme dilleri getirilen.)

Bu usulyýet maglumatlaryň uly birmeňzeş massiwlerini işlemek üçin netijeli ulanyp bilner. Şeýle massiwler hasaplaýyş we statistiki usullar amala aşyrylanda köp duş gelýärler. Şeýle hem parallel programmirleme usulyýeti operasion ulgamlarynda we hakyky wagt ulgamlarynda modelirmekde üstünlikli ulanylýarlar.

13. Logiki parallel programmirlemegiň usulyýeti

Logiki programmirleme ýeke-täk parallel amala aşyrylyşa rugsat berýär. $a:-b,c$. mysalda b we c maksatlary ylalaşdyrmaklygyň tertibiniň hiç hili tapawudy ýok, şonuň üçin olary parallel subut etmek bolýar. Bu ýagdaýda b we c subut etmek prosesleri prosesleriň WE-ulgamyny emele getirýärler diýilýär. Eger-de ulgama girýän her bir proses üstünlikli bolsa, WE-ulgam üstünlikli subut edilýär. Eger-de ulgamdaky prosesleriň iň bolmanda biri üstünlikli bolsa ýa-da ulgamy üstünlikli subut edilýär. Prosesleriň ulgamy üçin umumy üýtgeýänler ulgamlary prosesleriň arasyndaky akymlara özgerýärler.

Logiki parallel programmirlemegiň dilleri

Logiki parallel programmirlemegiň iň belli bolan dilleriň biri – bu Ehud Şapiro bilen hödürlenen Concurrent Prolog dili. Bu dil abstrakt spesifikasiýalara logiki çemeleşmäni ösdürýär we Deýkstranyň käbir taglymatlaryny öz içine alýar.

Concurrent Prolog dili obýekte-gönükdirilen programmirlemegiň dürli serişdelerini goldaýar. Emma olar unifikasiýa häsiýetlerine örän baglydyr. Obýekte gönükdirilen dillerinde habar maksatlaýyn obýektiň adyny görkezmek bilen iberilýär. Concurrent Prologda obýektler umumy üýtgeýänler bilen baglanyşykly we habarlar umumy üýtgeýänleriň görkezmeleri boýunça iberilýärler. Şonuň üçin ha-

bary ugratmak fazasynda maksatlaýyn obýektiň adyny görkezmek hökman däl.

Prosesleriň ulgamy düşünjesini girizeliň:

- prosesleriň WE-ulgamy – bu prosesleriň topary:
 - eger-de onuň ähli prosesleri üstünlikli amala aşyrylsa; üstünlikli amala aşyrylýar.
 - eger-de onuň prosesleriniň iň bolmanda biri şowsuz amala aşyrylsa, şowsuz amala aşyrylýar. Şunlukda, WE-ulgam özüniň ähli beýleki proseslerini zorluk bilen şowsuz amala aşyrýar;
 - eger-de onuň ähli prosesleri saklanyp galan bolsalar ýa-da tiz şowly tamamlansa saklanyp durýar;
- prosesleriň ÝA-DA-ulgamy – bu prosesleriň topary:
 - eger-de onuň ähli prosesleri şowsuz amala aşyrylsa; şowsuz amala aşyrylýar.
 - eger-de onuň prosesleriniň iň bolmanda biri üstünlikli amala aşyrylsa, üstünlikli amala aşyrylýar. Şunlukda, ÝA-DA-ulgam özüniň ähli beýleki proseslerini zorluk bilen şowsuz amala aşyrýar;
 - eger-de onuň ähli prosesleri saklanyp galan bolsalar ýa-da tizden-tiz şowsuz tamamlansa, saklanyp durýar.

III. PROGRAMMA ÜPJÜNÇİLIGINI DÖRETMEKLIĞIŇ TEHNOLOGIKI PROSESLERI



Programma üpjünçiligini döretmekliğin tehnologiýalarynyň esasy düşünjeleri

Programmanyň ýaşayyş sikli – bu programmany döretmek piki-rinden başlap, ony ulanmaklygyň görnüşleriniň ählisiniň tamamlan-magyna çenli döwürdir.

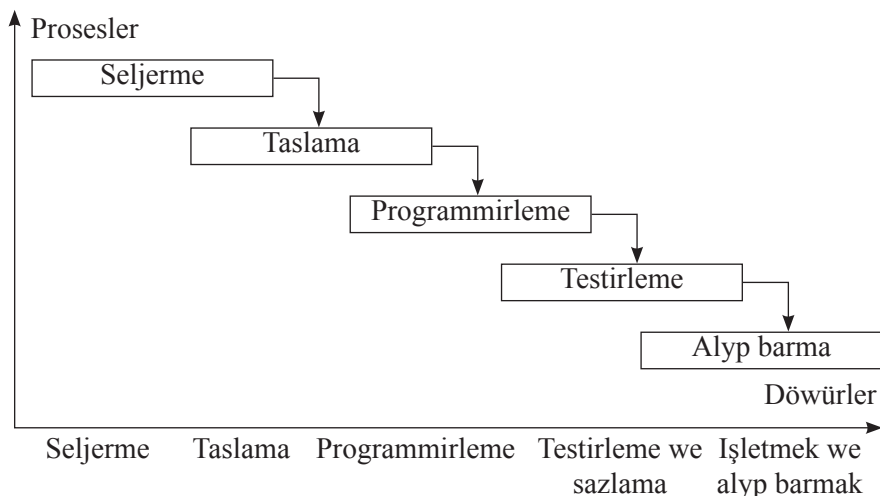
Programmirlemegiň tehnologiýasy tehnologiki proseslerini we olaryň geçiş tertibini, döwürlerini (bilimleri, usullary we serişdeleri ulanmak bilen) öwrenýär. Bilimler, usullar we serişdeler dürli proses-lerde, diýmek, tehnologiýalarynda ulanylyp bilner.

Programma üpjünçiligini döretmekliğin tehnologiýasy – pro-grammalary işläp düzmekliğin usullarynyň we serişdeleriniň toplu-myny, şeýle hem bu usullary we serişdeleri ulanmaklygyň tertibini saklaýan, programmalary işläp düzmekliğin ylmy we tejribede apro-birlenen strategiýasydyr.

Tehnologiýalary iki: dikleýin (prosesleri görkezýän) we kese-leýin (döwürleri görkezýän) ölçeglerde häsiýetlendirip bolar.

Proses – käbir giriş maglumatlary çykyş maglumatlaryna özgerd-ýän özara baglanyşykly amallaryň toplumydyr. Prosesler amallaryň jeminden, her bir amal bolsa, meseleleriň jeminden ybarat. Dikleýin ölçegler prosesleriň statik aspektlerini görkezýär we iş prosesleri, amallar, meseleler, amala aşyrylyşyň netijesi we ýerine ýetirijiler ýaly düşünjeleri peýdalanýar.

Döwür – bu käbir wagt çäkleri bilen çäklendirilen we berlen tapgyr üçin berlen talaplary bilen kesgitlenýän, anyk bir önümiň öndürilişi bilen tamamlanýan programma üpjünçiligini döretmek boýunça amallaryň bir bölegidir. Döwürler, adaty, iterasion häsiýe-te eýe bolan tapgyrlardan ybarat. Käwagt döwürleri fazalar diýip at-landyrylýan has uly wagt çäklerine birleşdirýärler. Şeýle keseleýin ölçeg wagty emele getirýär, prosesleriň dinamiki aspektlerini görkez-



14.1-nji surat. Ýaşayyş siklini alyp barmakda kaskad görnüşli tehnologiýa çemeleşme

ýar we fazalar, döwürler, tapgyrlar, iterasiýalar we barlag nokatlary ýaly düşüňjeler bilen amallary geçirýär.

Tehnologiýa çemeleşme programma üpjünçiliginiň dürli klaslaryna we işläp düzüjileriň toparynyň aýratynlyklaryna gönükdirilen, tapgyrlaryň we prosesleriň utgaşmalarynyň spesifikasiýasy bilen kesgitlenýär.

(Programmanyň ýaşayyş sikliniň ýönekeý görnüşi 14.1-nji suratda görkezilen.)

Bu ýagdaýda her tapgyrda ýeke-täk proses ýerine ýetirilýär. Uly programmalar döredilende bu shema ýeterlik derejede korrekt däl. Emma bu çyzygy beýleki köp tehnologiýa çemeleşmeleriň esasy hökmünde almak bolar.

15. Programmany işläp düzmekligiň umumy düzgünleri

Programma – bu önüm çylşyrymly obýekt we ol aşakdakylar bilen şertlendirilýär:

- meseläniň çylşyrymlylygy bilen;
- dolandyrylmagyň çylşyrymlylygy bilen;

- aýratyn bölek ulgamlaryň özüni alyp baryş beýanyň çylşyrymlylygy bilen;
- programma önüminiň aýratyn bölekleriniň çeyeligini üpjün etmekligiň çylşyrymlylygy bilen şertlendirilýär.

Programmany işläp düzmekligiň umumy düzgünleri aşakda getirilýär:

Ýygylýk düzgüni

Algoritmelerde we maglumatlarda ulanylyş ýygylýgy boýunça aýratyn toparlary bellemek. Programmanyň işinde has köp duş gelýän amallar üçin olaryň çalt ýerine ýetirilmegi üçin şertler döredilýär. Ýgy-ýgydan ulanylýan maglumatlar has çalt elýeterli üpjün edilýär. Belli bolşy ýaly, programmanyň operatorlarynyň diňe 5%-i programmanyň ýerine ýetiriliş tizligine täsir edýärler. Bu fakt programmanyň operatorlarynyň köpüsini hasaplaýyş tizligini hasaba alman, kodlamaga we esasan, programmanyň «owadanlygyna» we tekstleriň aýdyňlygyna esasy üns bermäge mümkinçilik berýär.

Modullyk düzgüni

Programmanyň düzüm böleklerini modullara özbaşdaklaşmagynyň usullary köp derejede programmany taslamak üçin ulanylýan usuly bilen kesgitlenýär.

Modul – bu beýleki moduldan çagyryp bolýan we özbaşdak kompilirlenýän ýapyk programma. Beýleki kesgitleme: *programma moduly* – prosesiň beýanlarynda ulanylyş üçin ýaramly bolan we özbaşdak programma önümi hökmünde taýýarlanýan prosesi beýan etmekligiň islendik bölegi.

Modul – üç sany esasy atributlara eýedir:

- ol bir ýa-da birnäçe funksiýalary ýerine ýetirýär;
- modul käbir logikany (algoritmi) amala aşyrýar;
- bir ýa-da birnäçe kontekstlerde ulanylýar.

Bu ýagdaýda esasy zat funksiýanyň nähili amala aşyrylýany däl-de, bu modulyň amala aşyryýan işidir. Logika bolsa modulyň öz funksiýalaryny nähili amala aşyryandygyny häsiýetlendirýär. Kontekst anyk ulanylyşyny beýan edýär.

Modullaýyn programmirlemegiň prinsipleri:

- Her modulda içki aragatnaşyklaryň güýçlendirilmegi (beýleki ýagdaýda bu düzgüne modulyň berkligini ýokarlandyrmak düzgüni diýilýär);
- Modullaryň arasyndaky arabaglanyşygy gowşatmak (beýleki ýagdaýda prinsipe modullaryň birleşmesini peseltmek düzgüni diýilýär).

Funksional seçip alyjylyk düzgüni

Programmada hasaplaýyş prosesiniň netijeli guramaklygy üçin hemişe taýýar ýagdaýda bolup durmaly, wajyp modullaryň käbir bölegi bellenilip alynýar. Programmanyň bu bölegine ýadro ýa-da monitor diýilýär. Monitoryň düzümine girýän programmalar hemişe operativ ýatda saklanylýarlar, galan modullar – goşmaça ýüklenýärler.

Generirlenme düzgüni

Deslapky programma tehniki serişdeleriň anyk konfigurasiýasy-na, çözülýän meseleleriň toparyna, ulanyjynyň iş şertlerine sazlanýan bolmaly.

Funksional artykmaçlygy düzgüni

Bu düzgüni şol bir işi dürli serişdeleriň kömegi bilen amala aşyrmak mümkinçiligini göz önünde tutýar (ulanyjynyň amatlylygy üçin).

«Agzalmadyk ýagdaýynda» düzgüni

Bu düzgüni gurluşlaryň, modullaryň, enjamlaryň we maglumatlaryň konfigurasiýalarynyň käbir bazalaýyn beýanlarynyň programma bilen işlemekligiň şertlerini kesgitlemek ulgamda ýatda saklanmagyna esaslanan.



16. Programma üpjünçiligini döretmekligiň tehnologi klassifikasiýasy

Tehnologiiki prosesleriň iki toparyna (köplüğine) seredýäris. Birinji topar – *klassiki*, ol programma üpjünçiligini işläp düzmekligiň tejribesiniň netijesinde dörän esasy proseslerini öz içine alýar. Ikinji topar – *standart*, ol ISO 12207:1995 standartyna esaslanan. Umuman,

klassiki toparyň prosesleri ýa-da prosesleriň amallary hökmünde, şeýle hem standart toparynyň bölegi bolup hyzmat edýärler.

Klassiki toparda dokuz sany tehnologiكي prosesleri bellemek bolar:

- ideýanyň döreýşi we derňewi;
- dolandyrylyşy;
- talaplaryň seljerilişi;
- taslanylyşy;
- programmirlenişi;
- testirlenmegi we sazlanmagy;
- işe girizilişi;
- ekspluatirlenişi we alyp barylýşy;
- eksplutasiýanyň tamamlanmagy.

ISO 12207 [ISO/IEC 12207:1995] halkara standarty bilen kesgitlenýän ýaşayş sikliniň prosesleri üç topara bölünýärler:

- esasy prosesler:
 - satyn almaklyk;
 - gowşurmaklyk;
 - işläp düzmeklik;
 - eksplutasiýa;
 - alyp barmaklyk;
- kömekçi prosesler:
 - resminamalaşdyrmaklyk;
 - konfigurasiýany dolandyrmaklyk;
 - hil bilen üpjün etmeklik;
 - werifikasiýa;
 - attestasiýa;
 - bilelikde bahalandyрма;
 - audit;
 - meseleleri çözmeklik;
- guramaçylyk prosesler:
 - dolandyrmaklyk;
 - infrastrukturany döretmeklik;
 - kämilleşdirmeklik;
 - okatmaklyk.



Programma önümini döretmek ideýasynyň döreýşi we onuň derňewi

Ideýanyň döreýşi we derňew prosesi, adaty, indiki üç tapgyrdan ybarat bolýar:

- ideýanyň döremegi we onuň maksimal döredijilige hem-de formal däl häsiýete eýe bolan ilkinji derňewi;
- ideýanyň jikme-jik derňewi, konsepsiýanyň işlenilmegi, meseläniň goýluşy, «taslamanyň bir sahypalaýyn beýanyny» döretmek we onuň giňeldilen görnüşini (wersiýasyny) – tehniki ýumşy işläp taýýarlamak. Buýruk esasynda döredilýän programma önümi ýagdaýynda tehniki ýumuş buýrujy tarapyndan taýýarlanylýar;
- ideýanyň hünärmenler tarapyndan amala aşyrylýan ekspertizasy ýerine ýetiriji bilen tehniki ýumşu ylalaşmak we meýilnamalaşdyryş prosesiniň başlanmagy barada çözgüt kabul etmek.

Ideýa eýýäm bar bolan programma önüminiň kämilleşdirilmegine ýa-da täzesiniň döredilmegine getirip biler.

Meseläni çözmek ideýasynyň döremegi

Berlen proses, adaty, adamda ýa-da adamlaryň bir toparynda meseläni çözmekligiň ideýasyny döremekden başlaýar we ol:

- programma önüminiň döredilmegine we kämilleşdirilmegine böwet bolýar;
- programma önümünde ýalňyşlyklara getirýär.

Olaryň içinde innowasion çözüwler has uly gyzyklanma bildirýärler. *Innowasiýa* – eýýäm bar bolan ulgamy, adaty, položitel netije bilen üýtgetmegi ýa-da käbir meseläniň täze çözüdini hödürleýän täze tehnologiýalar. Innowasiýalar oýlap tapmalary, açyşlary we ýönekeý täzelikleri hem öz içine alýar. Netijeli innowasiýa – bu eýýäm bar bolan çözüdi, onuň käbir häsiýetnamalaryny gowulandyryp üýtgedýän ýa-da ön çözülmelik meseläniň çözüdiniň položitel netijesini berýän innowasiýadyr. Netijeli innowasiýalaryň mysallary bolup, Internet global torunyň döredilmegi we integrirlenen programmirme gurşawynyň işlenilip döredilmegi çykyş edýärler.

Meseläniň goýluşy

Indi ideýa taslama öwrülmeli. Netijede, döredilýän programma önüminiň konsepsiyasy işlenilmeli. Onuň üçin taslamanyň mak-sadyny anyklamaly, taslama we onuň netijelerine talaplary kesgitle-meli. Bu maksala ýetmek üçin we talaplary ýerine ýetirmek üçin bar bolan çäklendirmeleri ýüze çykarmaly.

Taslamanyň bir sahypaly beýany öz içinde hödürlenýän ideýanyň dürli taraplaryny beýan edýän, birnäçe uly däl bölümleri saklaýar. Bu beýan hökman bir sahypadan däl-de, birnäçe sahypadan hem düzülip bilner. Emma ol iki, üç sahypadan köp bolmaly däl. Programma önümi üçin resminamalary işläp taýýarlamak bilen bagly bolan taslamanyň beýanyň mysalyny getireliň.

OpenMP boýunça gollanmanyň işlenilmegi

Gysgaça maglumatlar

OpenMP standarty direktiwalaryň toparyny emele getirýär we yzygider ýerine ýetirilişi üçin ýazylan koda olaryň goşulmagy kompilyatora parallel hasaplaýyş üçin programmany bölek meselelere bölmäge mümkinçilik berýär. OpenMP döremegi, programmalaryň parallel ýerine ýetirilmegi üçin ilkinji standart bolup durdy.

«OpenMP boýunça ulanyjynyň gollanmasy» resminamanyň işlenilip taýýarlanmagy hem göz önünde tutulýar.

Giriş

Taslamanyň ady: OpenMP boýunça ulanyjynyň gollanmasy.

Resminamany taýýarlamaklygyň senesi: 19-njy awgust 1998-nji ýyl, 1,3-nji görnüş.

Resminamanyň aýratynlyklarynyň beýany:

- «OpenMP boýunça ulanyjynyň gollanmasy» resminama Internet torundaky kompaniýanyň serwerinde hem kagyz görnüşinde hem elýeterli bolar;
- OpenMP spesifikasiýasynyň haýsy aýratynlyklarynyň amala aşyryljakdygy anyk görkezilmeli;
- OpenMP boýunça doly spesifikasiýa Internetden elýeterli (<http://www.openmp.org/>).

Resminamanyň ulanyjlary

Ulanýjy – C, C++, Pascal we FORTRAN dillerinde tejribe-li programma düzüji bolmaly we köp prosesorly kompýuterlerde

programmalaryň parallelleşdirilmeginiň artykmaçlyklaryna düşünmelidir.

Deňşdiriji seljeriş

Kompilýatorlary düzüjileriň köpüsi (mysal üçin, SGI, Sun, IBM kompaniýalaryň) OpenMP goldawyny eýýäm birikdirdiler. OpenMP beýanyny olar C, C++, Pascal we FORTRAN dilleriniň kompilýatorlary boýunça ulanyjynyň gollanmalaryna aýratyn bap görnüşinde goşýarlar.

Tehniki prosesiniň beýany

1. OpenMP spesifikasiýalaryny resminamadan importirlemek mümkinçiligini seljermek.

2. OpenMP goldawyny amala aşyrýan inženerleriň komegi bilen OpenMP spesifikasiýalarynyň haýsy aýratynlyklary kompilýatoryň şu görnüşine girjekdigini kesgitlemeli.

3. «Gollanmanyň» taslamasyny işläp taýýarlamak we ony ýaýratmaklygyň usulyny kesgitlemeli (marketing topary bilen bilelikde).

4. Amala aşyrmaklygyň detallaryny seljermek, «gollanma» degişli programmalaryň mysallaryny taýýarlamak we girizmek.

Esasy resminamalaryň sanawy

OpenMP spesifikasiýasy.

Esasy seneler

– «Gollanmany» taslamaklygy tamamlamak – aprel aýy, 2014-nji ýyl.

– «Gollanmanyň» doly gutarnykly ýagdaýdaky ilkiji görnüşini taýýarlamak – awgust aýy, 2014-nji ýyl.

Çeşmeler

Taslamany amala aşyrmak üçin zerur bolan çeşmeler 17.1-nji tablisada getirilen.

17.1-nji tablisa

Taslamany amala aşyrmak üçin zerur bolan çeşmeler

Funksiýa	Stawka	Düşündiriş
Inžener-düzüji	0,1	OpenMP amala aşyrýan maslahatçy inžener
Tehniki ýazyjy	1,0	Resminamany taýýarlamak
Taslamanyň ýolbaşçysy	0,3	Dolandyrmak, meýilnamany we mazmuny işläp taýýarlamak



Programma önüminiň taslamasyny dolandyryş prosesi

Programma üpjünçiliginiň ösen bazary we örän güýçli bäsdeşlik şertlerinde maksimal girdeji almak, işläp düzmek möhleti we programma düzüjileriň beýleki toparlary bilen ýakyn gatnaşykda bolmak ýaly kriteriler möhüm rol oýnaýarlar.

Programma önüminiň professional işlenilip taýýarlanylmagynda uly firmalarda we kompaniýalarda programmany düzmek prosesine goşmaça islegleri talap edýärler:

- programma önümini işläp düzmek üçin formalizirlenen modeliniň bolmaklygy;
- resurslaryň we ähmiýetliligiň in gowy ýerleşdirilmegi;
- anyk we resminamalaşdyrylan çözgütleri kabul etmekligi;
- köpsanly standartlary berjaý etmekligi;
- işläp düzüjileriň toparlarynyň arasynda netijeli aragatnaşygy üpjün etmekligi, şeýle toparlary çalt döretmek we olaryň iş prosesine tiz girmekligi;
- goldaw we satyjy toparlar bilen ýakyn aragatnaşykda bolmaklyk.

Dolandyryş prosesi programma önüminiň ähli ýaşaýyş siklinde dowam edýär. *Taslamanyň dolandyrylyşy* – bu berlen wagt, pul serişdelerinde, beýleki resurslardaky, şeýle hem taslamanyň gutarnykly netijesiniň hilinde çäklendirmelerde maksimal mümkin bolan netijeligi bilen taslamany amala aşyrmaklyga gönükdirilen işdir.

18.1. Taslamanyň dolandyrylyşynyň usullary

Taslamanyň dolandyrylyşynyň usullarynyň esasynda geçen asyryň 50-nji ýyllarynyň soňunda işlenilen tor meýilnamalaşdyrmaklygyň usullary ýatrylar. Bu usullaryň kömegi bilen taslamanyň ýolbaşçysy indikleri amala aşyryp biler:

- taslamanyň üstünde işi wagtyndan ön meýilnamalaşdyryp, kynçylyk döredip biljek we taslamany wagtynda ýerine ýetirmekligi saklap biljek sebäpleri önünden görmek;

- ýumuşlaryň ýerine ýetirilişiniň talap edilýän yzygiderliligine laýyklykda işleri wagtynda tamamlamak;
- kalendar grafigi saklamak we taslamany wagtynda tamamlamak üçin işleriň ýerine ýetirilişine gözegçilik etmek we koordinirmek.

Indiki iki usul parallel we biri-birine garaşsyz döredilenler:

- *kritiki ýoluň usuly (Critical Path Method – CPM)*. Bu usul «Dupont de Nemours» firma tarapyndan işlenilip taýýarlanyldy;
- *programmalary seljermek we bahalandyrmak usuly (Program Evaluation and Review Technique – PERT)*. Bu usul Lockheed korporasiýasy we «Booz_alen & Hamilton» konsalting firma tarapyndan işlenilip taýýarlanyldy.

Emma häzirki zaman taslamalaryny beýan etmek üçin, bu usullaryň serişdeleri ýeterlik däl. Tor meýilnamalaşdyryş usuly graflaryň nazaryýetiniň serişdeleriniň kömegi bilen işleriň baglanyşygyny beýan edýär.

18.2. Taslamanyň meýilnamalaşdyrylyşy

Taslamanyň meýilnamalaşdyrylyşy adam, hasaplaýyş we guraýyş resurslaryny kesgitlemekligi, şeýle hem meseleleriň we olary ýerine ýetirmek üçin wagtyň «kartasyny» düzmekligi öz içine alýar. Prosesleriniň standart toplumynda meýilnamalaşdyryş dolandyrylyş prosesiniň amallarynyň biri bolup durýar. Taslamanyň meýilnamalaşdyryşyna umumy çemeleşme şeýle basgançaklardan durýar:

- meseleleriň sanawyny düzmek;
- işleriň ýerine ýetirilişiniň grafiklerini gurmak;
- çykadjylary hasaplamak;
- talap edilýän resurslary bellemek;
- jogapkärçiligi paýlaşdyrmak;
- meseleleriň arasynda baglanyşyklary kesgitlemek;
- meselelere personal wezipeleri kesgitlemek;
- meseläniň ýerine ýetiriliş wagtyny kesgitlemek;
- anyk meseleler bilen bagly bolan howplary kesgitlemek;

- kritiki ýollary ýüze çykarmak;
- dolandyrylyşyň infrastrukturasyny döretmek.

Iş, onuň öňündäki ähli şertler ýerine ýetirilmese başlap bilmeýär. Ähli taslamanyň ýerine ýetiriliş wagty onuň ýolundaky ähli işleriň jeminden kiçi bolup bilmeýär. Ýoluň uzynlygy diýip ýola girýän işleriň dowamlylygynyň jemine aýdylýar. *Kritiki ýol* diýip iň uly uzynlykdaky ýola aýdylýar. Taslamanyň ýerine ýetirilişinde kritiki ýoldaky işleriň saklanmagy iň uly howpa eýe bolar. Ilki başda ýüze çykan kritiki ýol soňra beýleki kritiki ýol bilen çalşyrylyp bilner. Bu bolsa taslamanyň dolandyrylyşy boýunça kabul edilen çözgütleriň ähmiýetini ýitirdip biler.

18.3. Işleri paýlaşdyrmak

Işleriň paýlaşdyrylyşy meseleleri ýerine ýetirijiler üçin kwalifikasiýa derejesini kesgitlemekligi, taslamanyň potensial gatnaşyjylarynyň sanawyny we meseleleriň ýerine ýetirijilerini kesgitlemekligi öz içine alýar. Talap edilýän iş ýerlerine talap edilýän kwalifikasiýaly ýerine ýetirijileri goýmaklykda uly meseleler ýüze çykyp bilerler. Personal işleri bellemeklik üçin birnäçe düzgünler ulanylýar:

- kritiki ýolunyň has çylşyrymly meselelerine tejribeli we kwalifisirlenen adamlary bellemek. Az tejribeli adamlary çylşyrymlylygy boýunça ýeňilräk meselelere bellemek gerek;
- her bir ýerine ýetiriji üçin işleriň yzygiderligini kesgitlemek;
- ýerine ýetirijileri işlemek üçin has köp wagty talap edýän meseleler bilen ýüklemeli däl.



19. Talaplaryň seljerilişi we programma önüminiň taslanylyşy

19.1. Talaplaryň seljerilişi, programma önüminiň taslanylyşy we arhitekturasy

Seljeriş we taslanyş – iki sany ýakyn we örän baglanyşykly düşüňjedir. Olar bir umumy meseläni çözüýärler. Onuň netijesinde ulgamyň anyk görnüşi emele gelmeli hem-de şonuň esasynda pro-



gramma kody düzülmeli. Şonuň üçin biz olara bilelikde seredýäris hem-de usulyýetler, düzümläýin usulyýet, seljerişe, taslanyşa, obýekte gönükdirilen usulyýetler boýunça seljerişe we taslanyşa böleris.

Talaplaryň seljerilişi – bu programmanyň ýaşaýyş sikliniň prosesi we onuň dowamynda buýrujynyň talaplary anyklanylýar, formalizirlenýär hem-de resminamalaşdyrylýar. Bu ýerde «Dörediljek programma näme etmeli?» diýen esasy sorag döreýär.

Taslanyş – bu hem programmanyň ýaşaýyş sikliniň prosesi we onuň dowamynda programmanyň, gurluşynyň elementleriň özaragatnaşyklarynyň derňewi geçirilýär. Taslanyş düzgün bolşy ýaly, iterasion proses bolup durýar. Bu ýerde döreýän esasy sorag «ulgam alnan talaplara nädip jogap berer?».

Taslanyş iki derejede geçirilmeli:

- arhitekturanyň taslanylyşy (ulgamy taslamak, «uludan» taslamak);
- bölekleyin taslanyş maglumatlary (modullaryň taslanylyşy, «kiçiden» taslamak).

Seljerişniň we taslanyşyň netijesi bolup, onuň esasynda ulgamy amala aşyrmak üçin ýeterlik maglumatlary özünde saklaýan taslama çykyş etmeli. Şu günki günde programmirlemede iki sany esasy düzümläýin we obýekte gönükdirilen usulyýet ulanylýar.

Spesifikasiýalar – bu käbir ýerine ýetirilmeklige degişli bolan işiň jikme-jik beýan edilişi. *Spesifikasiýanyň serişdeleri*: şeýle beýanlary düzmek ýa-da almak üçin islendik serişdeler. *Spesifikasiýanyň dili*: bu rasional taýýarlanylýan we sintaksis taýdan guramaçylykly bolan şeýle serişdeleriň toplumydyr.

Programma önüminiň arhitekturasy – onuň düzülişi daşyndan görnüşü ýalydyr, diýmek, programma önüminiň, birnäçe özara baglanyşykly bölek ulgamlardan ybarat bolan ulgam hökmünde görkezilýär. Şeýle bölek ulgamlar hökmünde aýratyn programmalar çykyş edýärler.

Programma arhitekturasyyny umuman beýan edýän dört gurluş kesgitlemek bolar:

- *logiki (konseptual) gurluş*. Ol ulgama abstrakt derejede funksional talaplary beýan etmek üçin zerur bolan abstraksiýalaryň köplüginde öz içine alýar;

- *modullaýyn gurluş*. Ol programma serişdesini modullaryň top-
luny ýa-da taslama toparynyň agzalary bilen ýerine ýetiril-
ýän iş ýumuşlaryny deňeşdirip bolýan programma birlikleri
hökmünde kesgitleýär;
- *prosesleýin gurluş*. Ol ulgamyň, onuň ýerine ýetirilişiniň do-
wamynda özüni alyp barşyny beýan edýär;
- *fiziki gurluş*. Ol programma serişdesiniň elementlerini apparat
üpjünçiligine geçirilişini kesgitleýär.

Bu dört gurluşlardan başga beýleki gurluşlar hem ýygy-ýygydan
ulanylýarlar:

- çagyryşlaryň gurluşy;
- maglumat akymlarynyň gurluşy;
- dolandyryş akymlarynyň gurluşy;
- ulanyş gurluşy;
- dynplaryň gurluşy.

19.2. Arhitekturanyň taslanylyşy («uludan» taslamak)

Düzümleýin usulyýeti

Düzümleýin usulyýet üçin arhitekturanyň taslanylyşy («uludan»
taslamak) indiki esasy usullary öz içine alýar:

- aşaklygyna taslanyş usuly;
- ýokarlygyna taslanyş usuly;
- özeni giňeltmek usuly.

Düzümleýim usulyýet ýagdaýynda programmanyň modullaýyn
strukturasy hökmünde agaç görnüşli gurluşy ulanmak kabul edilen.
Şeýle agajyň düwünlerinde programma modullary ýerleşdirilýär, ugruk-
dyrylan dugalar bolsa modullaryň statistiki baglylygyny görkezýärler.

Aşaklygyna we ýokarlygyna taslanyş usullarynyň örän köp
görnüşleri we modifikasiýalary bar.

Aşaklygyna taslanyş usuly aşakda getirilýän iki strategiýanyň
esasyndaky funksional dekompozisiýanyň çemeleşmesini emele ge-
tirýär:

- *ädimleýin anyklamak strategiýasy* boýunça dekompozisiýa-
nyň her bir soňky tapgyrynda ondan aşakda ýerleşen modullar
kesgitlenýärler;

- *habarlary seljermek strategiýasy* boýunça modullar bilen işlenilýän maglumatlaryň akymlary seljerilýärler.

Ýokarlygyna taslanyş usuly programmany taslamaklyk ilki bilen zerur boljak kömekçi modullary kesgitleýän çemeleşmäni emele getirýär.

Özeni giňeltmek usuly – bu esasy ünsi umumy programmanyň fuksiýalaryny kesgitlemeklige däl-de, kömekçi modullaryň köplügini ýüze çykarmaklyga berýän çemeleşmedir.

Obýekte gönükdirilen usulyýet

Obýekte gönükdirilen usulyýet üçin arhitekturanyň taslanylyşy-ny indiki esasy usullary:

- predmet ugurlaryny taslamaklygyň usuly;
- köprüleri gönükdirmek usuly öz içine alýar.

Predmet ugurlaryny taslamaklygyň usuly ulanyjynyň nukdaýnazaryndan ulgamyň predmet ugurlaryny bellemekden ybarat. Predmet ugry – bu özüni domen üçin häsiýetli düzgünlere we häsiýet çyzyklara laýyklykda alyp barýan anyk obýektleriň toparý bilen doldurylan hakyky, gipotetik ýa-da abstrakt dünýäsi.

Köprüleri gönükdirmek usuly boýunça bir predmet ugry beýleki predmet ugrunyň üpjün edýän mümkinçiliklerini we mehanizmlerini ulanýar. Iki predmet ugurlarynyň arasyndaky köpri teklipleriň toparýny (ulanyjynyň nukdaý nazaryndan) we talaplaryň toparýny (ýerine ýetirijiniň nukdaýnazaryndan) emele getirýär.

19.3. Modullaryň taslanylyşy («kiçiden» taslamak)

Düzümleýin usulyýet

Düzümleýin usulyýet üçin modullary taslamaklygyň esasy usullaryny sanap geçeliň we olaryň käbirlerine düşünje bereliň:

- «Esas-aragatnaşyk» diagrammalary;
- düzümleýin kartalar;
- işleýşiň diagrammalary;
- Warnýe-Ornyň diagrammasy;
- ýagdaýlaryň geçelgeleriniň diagrammalary;
- blok-shemalar;
- psewdokod.

«Esas-aragatnaşyk» diagrammasy (Entity-Relation Diagram – ERD) maglumat modellerini işlemek üçin niýetlenen we maglumatlaryň hem-de olaryň arasyndaky aragatnaşyklarynyň kesgitlenişiniň standart usulyny üpjün edýär.

Işlenilip taýýarlanylýan «esas-aragatnaşyk» görnüşli diagrammalary tejribede hemişe diýen ýaly, diňe unar we binar aragatnaşyklara ýol berýän relýasion modele özgertmek gerek.

Pseudokod (pseudocode) – programmanyň dolandyrylyşynyň gurluşyny beýan etmek üçin arassa hakyky dilini ulanmak usuly. Şeýle dilleriň konstruksiýalary bloklaýyn, düzümläýin dilleriniň konstruksiýasyna ýakyn. Pseudokod kompýuteriň ýerine ýetirip biljek sözlemlerinden düzülen. Emma, olarda ulanylýan programmirlenme diline bagly bolan açar sözleriniň ýerine erkin görnüşdäki sözlemler ulanylýar.

Obýekte gönükdirilen usulyýet

Obýekte gönükdirilen usulyýet üçin modullary taslamaklygyň esasy usullaryny sanap geçeliň we olara häsiýetnama bereliň:

- kooperasiýanyň diagrammalary;
- komponentleriň diagrammalary;
- ýaýbaňlaşdyrmagyň diagrammalary.

Kooperasiýanyň diagrammasy (collaboration diagrams) özara baglanyşykda bolan obýektleriň arasyndaky ähli düzümläýin gatnaşyklary grafiki görnüşde görkezmek üçin niýetlenen. Kooperasiýa bellibir maksat bilen arabaglanyşykda bolan obýektleriň köplügin bellemek üçin ulanylýar. Kooperasiýa iki derejede görkezilip bilner:

- *spesifikasiýanyň derejesinde*. Seredilýän arabaglanyşykda klassifikatorlaryň we assosiasiýalaryň rollaryny görkezýär;
- *mysallaryň derejesinde*. Kooperasiýada aýratyn rollary emele getirýän ekzemplýarlary we aragatnaşyklary görkezýär.

Diagrammadaky obýektler grafyň depelerinde ýerleşýärler. Aragatnaşyklar depeleri dugalar bilen birikdirýärler. Aragatnaşyklar obýektleriň kabul edýän we ugradýan habarlary bilen annotirlenip bilinerler.

Komponentleriň diagrammasy (component diagram) ulgamyň fiziki aýratynlyklaryny beýan edýär. Ol komponentleriň arasyndaky

baglanyşyklary gurup, işlenilýän ulgamyň arhitekturasyny kesgitlemäge mümkinçilik berýär.

Ýaýbaňlaşdyrmagyň (ulanmaklygyň, ýerleşdirmekligiň) diagrammasy (deployment diagram) ulgamyň programma we aparat komponentalarynyň arasynda fiziki aragatnaşyklary görkezýär.



Programma önüminiň programmirlenilmegi (amala aşyrylyşy)

Programma düzmekligiň prosesinde döreýän möhüm tehnologiكي mesele – bir umumy programmirleme görnüşe laýyk getirmek. Programmirleme stili diýip dogry, netijeli, ulanyş üçin amatly we aňsatlyk bilen okalýan programmalary almak üçin tejribeli programma düzüjileriň ulanýan usullarynyň we ýollarynyň toplumyna düşünilýär. Adatça, programmirlemegiň oňat görnüşiniň düzgünleri tejribäniň netijesinde döreýän programma düzüjileriň arasyndaky ylalaşyklaryň netijesidir. Programma kody ýönekeý we düşnükli bolup, indiki häsiýetlere eýe bolmaly:

- aýdyň logika;
- adaty aňlatmalar;
- programmirleme dilinde kabul edilen ylalaşyklary ulanmak;
- manyly atlar;
- tertipli formatirleme;
- adaty däl, çylşyrymly we düşnüksiz konstruksiýalaryň bolmazlygy.

Uly taslamalaryň köpüsinde programmany işläp düzüjileriň toparynyň programmirleme stilini kesgitleýän içki resminamalary bolýar. Mysal üçin, GNU (http://www.gnu.org/prep/standarts_toc.html) taslamalarynyň programmirleme görnüşini boýunça resminama. Şeýle resminamalaryň ulanylmagy bir umumy görnüşini ulanmaklyga, netijede, programmalaryň okaýjylygyny we alnyp barlyşyny ýokarlandyrmaklyga, köp ýagdaýlarda-da ýalňyşlyklaryň önümini almaklyga getirýär.

Programmirleme stilleriniň iki görnüşlerine seredip geçeliň:

Gorag programmirlеме görnüşü;

Gorag programmirlеме (defensive programming) – bu programma düzülende emele gelyän ýalňyşlyklar programma düzüji bilen aňsatlyk bilen tapylyp identifikirlenýän görnüşidir.

Gorag programmirlемегің üç sany esasy düzgüni bar:

- *umumy ynamsyzlyk*. Her bir modul üçin giriş maglumatlaryny ýalňyş bolup bilerler diýen pikir bilen jikme-jik seljermeli;
- *bada-bat gözläp tapmaklyk*. Her bir ýalňyş mümkin boldugyça ýüze çykarylmalı, bu onuň sebäbini anyklamaklygy aňsatlaşdyrýar;
- *ýalňyşlary izolirlemek*. Bir moduldaky ýalňyşlyklar olaryň beýleki modullara täsiri bolmaz ýaly izolirlenen bolmalı.

Umumy ynammazlyk düzgüni dürli usullar bilen goldanyp bilner. C diliniň kitaphanasynda assert (aňlatma) makrokesgitleme bar we ol programma diagnostiki maglumatlary girizmek üçin ulanylýar. Eger aňlatmanyň bahasy nula deň bolsa, onda indiki görnüşdäki ýazgy ýüze çykar:

Assertion failed: aňlatma, file faýlyň_ady, line nnn

Ondan soň hasaplamalary tamamlayan kitaphana funksiýasy çagyrylar.

Indi gorag programmirlеме boýunça birnäçe maslahatlary bereliň:

- üýtgeýän ululyklaryň bahalar ýaýlasynyň barlagyny geçirmeli;
- üýtgeýän ululyklaryň bahalarynyň hakykata ýakynlygynyň barlagyny geçirmeli. Bu bahalar käbir konstantalardan ýa-da beýleki üýtgeýän ululyklaryň bahalaryndan uly bolmaly däl-dirler;
- hasaplaşyklaryň netijelerine gözegçilik etmeli;
- awtomatiki barlaglary işletmeli (mysal üçin, çenden aşalyga ýa-da takyklygy ýitirmeklige gözegçilik etmek);
- maglumatlaryň elementleriniň uzynlyklaryny barlamaly;
- funksiýalaryň gaýtaryş kodlaryny barlamaly.

Edebi programmirlеме görnüşü (literate programming)

Edebi programmirlеме görnüşü (literate programming) – bu programma kompýuter üçin instruksiýalaryň yzygiderligi däl-de, edebi eser hökmünde seredilýän programma ýazmaklygyň stili. Şeýlelikde,

programma düzüji programmany ýazanda, kompýuterde ýazy-
lan amallary, programmirlеме dilinde ýazylan kody degişli ýerlere
goýup, formal beýan edýär.

Edebi programmirlемegiň konsepsiýasy Donald Knut tara-
pyndan 1984-nji ýylda makala görnüşinde hödürlenildi. Bu makalada
programmalary resminamalaşdyrmaklygyň täze usuly we bu usuly
amala aşyran – «Kerep» – **WEB** diýip atlandyrylan ulgam bara-
da beýan edildi. Görnüşi amala aşyrmak üçin **WEB** ulgamy şol bir
wagtyň özünde programmanyň tekstiniň üstünde hem-de ony işläp
taýýarlamaklygyň resminamalary bilen işlemekligi hem hödürle-
di. Programmirlеме dili hökmünde C dilini ulanmaklyga adaptir-
lenen CWEB ulgamy has giňden ýaýrady. Bu ulgam (<http://www.computerra.ru/offline/2001/387/7737/>) programma düzüjiniň pro-
grammany «seksiýalara» bölmekligiň üsti bilen öz işini strukturir-
leýändigini göz önünde tutýar. Bu seksiyalara «Modullar», «Bölek
programmalar», «Funksiýalar» ýa-da ýöne koduň fragmentleri degişli
bolup bilerler, özi hem her bir seksiya üç bölekden ybarat:

- TeX-bölegi. Ol işläp düzüjiniň resminamalaryny we seksiyanyň
düşündiriş düzümini saklaýar;
- aralyk bölegi. Ol seksiya üçin spesifiki makro kesgitlemeleri
saklaýar;
- C-bölegi. Ol programma koduny saklaýar.

C we C++ dillerinde programmalary ýazmak görnüşi boýunça te-
matiki toparlara ýygnap, käbir maslahatlary getireliň. Bu maslahatla-
ryň köpüsini beýleki programmirlеме dillerinde hem şowly ulanmak
bolar.

Faýllaryň gurluşy

Faýllary atlandyrmak boýunça maslahatlary getireliň:

- sözbaşy faýllar *h* giňeltmeli bolmaly. C dilinde programmaly
faýllar *c* giňeltmeli, C++ dilindäki programmaly faýllar *cc*
(Unix operasion ulgamyna) ýa-da *cpp* (Windows operasion ul-
gamynda) giňeltmeli bolmaly;
- faýllaryň atlary eýýam ilkinji sekiz simwollar bilen tapawut-
lanmaly. Sebäbi käbiri könelişen hem bolsa-da, giňden ula-
nylýan operasion ulgamlary faýllaryň atlarynyň uzynlyklaryna
çäklendirmeleri goýýarlar;

- ähli faýllaryň atlary dürli papkalarda ýerleşen bolsalar-da dürli bolmaly.

Setirleriň uzynlyklary boýunça maslahatlary aşakdaky görnüşde kabul edeliň:

- setirler 80 simwoldan uzyn bolmaly däl. Şonuň üçin tekst redaktorlaryň we programmalaryň tekstlerini çap etmek ulgamlarynyň köpüsi setirde simwollaryň şu maksimal sanyna gönükdirilen. Bu çäklendirme ekranyň ýüzünde her biri 80 simwoldan ybarat 25 setiriň sygýanlygy bilen düşündirilýär;

- setiriň uzynlygy 80 simwoldan esli kiçi bolsa gowy.

Aşakda faýllaryň guramaklygy boýunça maslahatlar berilýär:

- C++ dilindäki programmanyň sözbaşy faýllarynda indiki tertibi saklamaly:

- blok üçin düşündiriş ýaly ýazylýan faýla düşündiriş. Düşündiriş berlen faýla hususylyk hukugyny kesgitleýän lisenziýa görkezme bilen başlanyp biler. Düşündiriş, adaty, faýllaryň wersiýalary bilen dolandyrmak ulgamynyň identifikasion setiri bilen dowam edýär. Mysal üçin, SCCS utilita üçin setir şeýle görnüşde bolup biler:

```
%Z%%M %I% %E%
```

we SCCS identifikasiýasynyň simwollaryny, faýlyň adyny, wersiýanyň nomerini we soňky modifikasiýasynyň senesi saklap biler. Düşündiriş faýlyň gysgaça beýanyny saklamaly;

- berlen faýlyň diňe bir gezek kompiliýasiýasyny üpjün edýän gorag barlagy. Ol şeýle görnüşde ýazylan bolmaly:

```
#if !defined(FILENAME_H)
```

```
#define FILENAME_H;
```

- importirlenýän interfeýsler. Düzgün bolşy ýaly, olar `#include`-derektiwalaryň kömegi bilen kesgitlenilýärler;

- yglan edilişler;

- gorag barlagy tamamlýan şertli kompiliýasiýanyň direktiwasy
#endif;

- C++ dilindäki kodly faýllar üçin şeýle yzygiderlik maslahat berilýär:

- faýla düşündirişler;

- importirlenýän interfeýsler;
- ýerli yglan edişler;
- eksportirlenýän kesgitlemeler.

Leksiki ylalaşyklar

Identifikatorlary ýazmak boýunça maslahatlary getireliň:

- identifikatory «_» alamaty bilen başlamaly-da gutarmaly-da däl;
- diňe baş harplardan düzülen identifikatorlary ulanmaly däl (Konstantalaryň we makro kesgitlemeleriniň atlary üçin degişli däl.);
- şol bir programmada şol bir harplaryň ýazylyşy bilen tapawutlanýan baş ýa-da setir harplardan düzülen atlary ulanmaly däl;
- identifikatoryň böleklerini «_» alamaty bilen bölmeli;
- identifikatorlar baş harpdan başlamaly.

Atlandyrmak boýunça ylalaşyklary beýan edeliň:

- ähli atlaryň manysy programma okalanda düşnükli bolmaly. Üýtgeýaniň ady (köp ýagdaýlarda) onuň niýetlendirilişini doly açyp görkezmeli;
- uzyn atlar gysgaldyp ýazylyp bilner. Möhüm däl suffiksleri ýok etmek ýa-da birinjiden başga ähli çekimli harplary ýok etmek ýa-da sözüň ilkinji harplaryny ulanmak ýaly düzgünlere laýyk hereket etmeli. Gysgaltmalaryň şol bir usullaryny ulanmak, şeýle hem ähli gysgaltmalary ýörite tablisalarda resminamalaşdyrmak maslahat berilýar (mysal üçin, program-maly faýlyň başynda);
- funksiýalaryň atlary aktiw bolmaly.

Düşündirişleri ulanmak boýunça maslahatlary getireliň:

- bloklaýyn düşündirişler faýl, synp, funksiýa ýa-da üýtgeýän barada goşmaça maglumatlary beýan etmek üçin ulanmaly;
- setirleýin düşündirişleri bir-iki setirli gysgaça belgijikler üçin ulanmaly.

Dil boýunça ylalaşyklar

Aňlatmalary ýazmak boýunça maslahatlary getireliň:

- binar operatorlar operandlardan bir probel bilen bölünen bolmaly;

- unar operatorlary, eger-de ony diliň düzgünleri talap etmeýän bolsalar, operanddan bölmek maslahat berilmeýär;
- okajylygy gowulandyrmak üçin ýaýlary ulanmaly. Girizilen ýaýlara probelleri goşmaly däl;
- aňlatmany bir setirde ýerleşdirmek maslahat berilýär;
- eger aňlatma indiki setire geçýän bolsa, onda binar amalyň alamatyny birinji setirde goýmaly.

Üýtgeýänleri yglan etmek boýunça maslahatlary getireliň:

- her bir yglan edişde diňe bir üýtgeýän ululyk yglan edilmeli;
- üýtgeýän ululygyň yglan edişi diňe bir setirde bolmaly;
- üýtgeýäniň görnüşinden soň probel, üýtgeýäniň ady we üýtgeýäni gysgaça beýan edýän düşündiriş bolmaly;
- eger üýtgeýän görkeziji bolsa, onda görnüşiniň yzyna «*» simwoly goşmaly.

Funksiýanyň kesgitlenilişiniň ýazylyşy boýunça maslahatlary getireliň:

- funksiýanyň kesgitlenilişi funksiýanyň sözbaşy bilen başlamaly, onuň yzyndan bolsa funksiýanyň niýetlenilişini beýan edýän düşündiriş we funksiýanyň özi ýazylmaly;
- funksiýanyň özüni öz içine alýan açylýan we ýapylýan ýaýlar setirleriň birinji sütüninde ýerleşýärler;
- sözbaşy funksiýa bilen gaýtarylýan bahanyň görnüşinden, probelden, funksiýanyň adyndan düzülýär;
- eger funksiýa statiki bolsa, onda *static* sözi funksiýanyň sözbaşyndan öň gelýän setire geçirilen bolmaly;
- funksiýanyň kesgitlenilişi faýlyň iň çepki sütüninde başlaýar;
- funksiýanyň dürli böleklerini boş setir bilen bölmeli.

Şertli operatory ýazmak boýunça maslahatlary getireliň:

- şertiň ýaýlarynyň önünden we soňundan bir probelden goýmaly;
- açylýan figura ýaýy ýapylýan tegelek ýaýyň setirinde bolmaly;
- ýapylýan figura ýaýy şertli operatoryň başlanýan yeriniň pozisiýasynda bolmaly;
- eger şert bir setire sygmaýan bolsa, onda onuň indiki setirdäki dowamy birinji setiriniň aňlatmasynyň birinji simwolyň ýerleşýän sütüninde başlamaly;

- eger *then* ýa-da *else* bölekleri ýönekeý operatoryndan ybarat bolsa, onda açylýan we ýapylýan figura ýaýlary goýulman hem bilerler.

Şertli operatoryň formatirlenilişi indiki programmada getirilýär.

Şertli operatoryň formatirlenilişiniň programmasy

```
If (some_result==NULL && some_calculation() ) {  
do_something ();  
and_some_more ();  
}  
else if (some_value>main_result ) {  
do_something_else ();  
}  
else {  
do_something_absolute_else ();  
}
```

Saýlama operatory ýazmak boýunça maslahatlar barada aýtsak, bu ýagdaýda esasy ylalaşyklar şertli operator üçin ylalaşyklar bilen meňzeş.

Saýlama operatoryň formatirlenilişiniň programmasy

```
switch (i) {  
case 1:  
do_something ();  
break;  
case 2: {  
do_something ();  
and_some_more ();  
break;  
}  
default:  
assert (false);  
break;  
}
```

Sikl operatorlary ýazmak boýunça esasy ylalaşyklar barada maslahatlar ýene-de şertli operatora meňzeşdir. Sikl operatory formatirlmekligiň mysallary aşakdaky programmada görmek bolar.

Sikl operatorň formatirlenilişiniň programmasy

```
for ( i=some_initial_value ();  
i>=some_other_value ();  
i=some_new_value () ) {  
do_something ();  
and_some_more ();  
}  
do {  
do_something ();  
and_some_more ();  
} while ( we_should_do () );
```

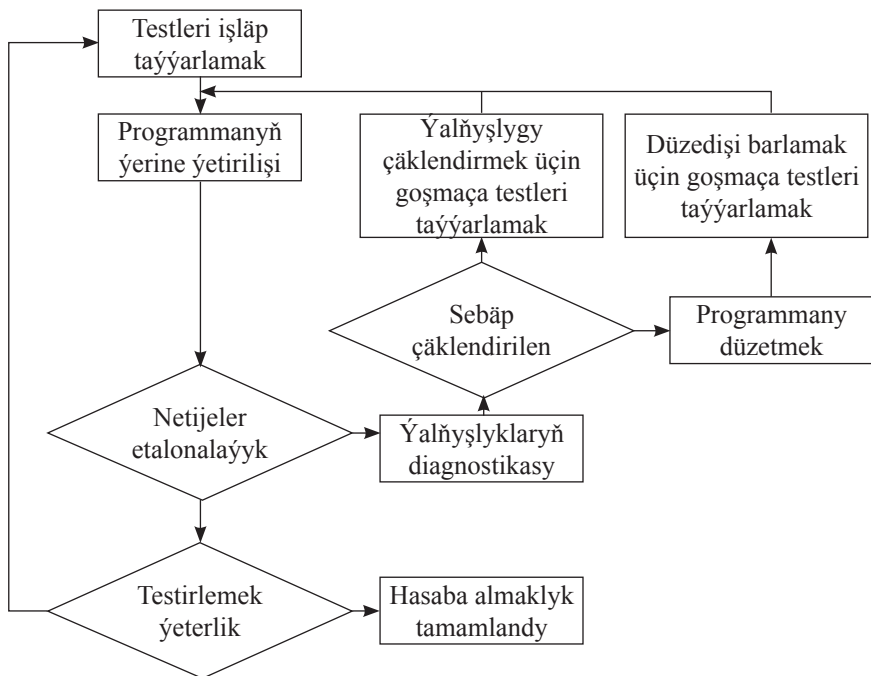


Programma önümlerini testirmek we ýola goýmak

21.1. Programma önümleriniň testirlenilişi

Testirleme – ýalňyşlary tapmak maksady bilen programmanyň ýerine ýetiriliş prosesi. Bu testirlemegiň klassiki kesgitlemesi Glenford Maýers tarapyndan berlen. Bu kesgitlemede testirleme prosesi ni diňe bir usulda görkezilen. Elde testirleme usullary (mysal üçin, programmalaryň inspeksiýalary we içinden görmek) hem bar.

Testirleme köp görnüşli testleri taýýarlamakdan we saýlanan usulyýetleriň birine laýyklykda olary barlamakdan başlanýar. Netijeleriň dogrulygy barlanandan soň, jogaplar deň gelmedik ýagdaýynda ýüze çykan meseläniň barlagy ýa-da jogaplar deň bolan ýagdaýynda testirlemekligiň ýeterlik derejede dolulygyny barlamak başlanýar. Testirmek doly derejede geçirilmedik bolsa ýa-da bar bolan testleriň kömegi bilen ýüze çykan ýalňyşlygy çäklendirip (lokallaşdyrmak) bolman, girizilen düzedişi barlap bolmaýan bolsa,



21.1-nji surat. Testirleme we ýola goýma

onda goşmaça testleri taýýarlamak gerek bolar (<http://www.testing-faqs.org/>).

Testirmekligiň iki sany esasy usuly bar:

- *programmanyň funksional testirlenmegi*. Bu usul boýunça programma içki düzümi belli bolmadyk obýekt hökmünde se-redilýär. Bu usul «gara ýaşşik» usuly at bilen hem belli. Bu usul senagat masşabynda iň giňden ýaýran we meşhur hasap edilýär;
- *programmanyň düzümläýin testirlenmegi*. Bu usul programmanyň içki koduny we oňa doly elýeterligi göz önünde tutýar. Bu usul «ak ýaşşik» usuly at bilen hem belli.

Testirmekligi testirlenýän birliklere görä hem tapawutlandyrylar. Mysal üçin, *proseduralaryň we funksiýalaryň testirlenilişi*, *modullaryň testirlenilişi*, *ulgamlaryň we bölek ulgamlaryň testirlenilişi*, şeýle hem *integrasiýa testirleme*.

Kompýuteriň kömegi bilen testirmeklige *dinamiki testirmek* diýilýär. Ol gurallaryň kömegi bilen amala aşyrylýan *awtomatiki tes-*

tirlemeklige (mysal üçin, test monitoryň kömegi bilen) we adam bilen amala aşyrylýan *elde testirmeklige* bölünýär. Elde testirmeklik ulanyjynyň grafiki interfeýsini testirmek üçin şu wagta çenli hem ulanylýar. Statiki testirmek testirlenýän programmany ulanmazdan, ony hünärmenler tarapyndan seljermekligiň netijesinde geçirilýär.

Islenilýän dolulygyna baglylykda testirmekligiň birnäçe görnüşlerini tapawutlandyrýarlar, olaryň arasynda birnäçelerine seredip geçeliň. *Determinirlenen testirleme* başlangyç etalon maglumatlaryň her bir kombinasiýasyny we programmanyň işiniň netijesinde oňa degişli alnan netijäniň kombinasiýalaryny barlaýar. Tejribede doly determinirlenen testirmekligi amala aşyrmak mümkin däl. *Tötänleýin (stohastiki) testirleme* statiki paýlaşdyrmak usulyny ulanmak bilen tötänleýin taýýarlanylýan başlangyç tekst maglumatlary ulanylýar.

Umuman aýdanymyzda, testirmekligiň örän köp kriterileriniň barlygy sebäpli, testirmekligiň görnüşleri örän köpdür.

Görnüşli ýalňyşlar

Gorag programmirleme stili görnüşli ýalňyşlaryň köpüsiniň önüni almaga mümkinçilik berýär. Programma düzüjileriň has giňden ýaýran ýalňyşlarynyň iki kategoriýasyna seredip geçeliň:

- programmada sintaksis barlagy geçirilenden soň hem galýan umumy häsiýetli (sintaksis däl) ýalňyşlar:
 - logiki ýalňyşlaryň topary. Mysal üçin, mümkin bolan şertleriň doly däl hasaby ýa-da şerti barlamakdan soň algoritmiň şahasyny nädogry görkezmek;
 - siklerde ýalňyşlar. Mysal üçin, sikliň başynyň we soňunyň nädogry çäkleri;
 - maglumatlar bilen işlenende ýalňyşlar;
 - üýtgeýän ululyk yglan edilende ýalňyşlar. Mysal üçin, üýtgeýän ululygyň inisializasiýasynyň ýoklugy;
 - massiw bilen işlemekde ýalňyşlar;
 - elementleriň başlangyç nullanmagynyň ýoklugy;
 - barlamak üçin has kyn bolan ýörite görnüşli ýalňyşlar;
 - parallel programmalar ýazylanda ýüze çykýan ýalňyşlar;
 - preprocessor ulanmak bilen bagly bolan ýalňyşlar;
 - amallaryň nädogry netijesi bilen bagly bolan ýalňyşlar;
 - ýityýan ýalňyşlar.

Programmalary işläp düzmek prosesinde testirmek

Işläp düzmek prosesinde entek ýazylyp gutarylmaýyk programmany testirmekli bolansoň, testirmeklige ähli çemeleşmeler iki topara bölünýärler:

- *ýokardan aşaklygyna testirmek.* Bu çemeleşme programma ýokardan aşaklygyna programmirlenende ulanylýar. Bu ýagdaýda «gapaklar» – programmanyň entek ýazyлмадык böleklerini imitirleýän koduň fragmentleridir;
- *aşakdan ýokarlygyna testirmek.* Bu çemeleşmede düzgün bolşy ýaly, uly däl programma – «draýwer» döredilmeli we ol eýýäm ýazylan modullaryň aragatnaşygyny gurnaýar.

Ýalňyşlyklary yzarlamak

Ýalňyş ýa-da beýleki ýüze çykan mesele ulgamda sekiz ýagdaýyň birinde bolup bilýän obýekt ýaly seredilip bilner. Bu ýagdaýlar indikilerdir:

- ýalňyş ýa-da beýleki ýüze çykan mesele hasaba alnan;
- ilkinji seljerme geçirildi (ekspert ýalňyşyň barlygyny tassyklady);
- ýalňyşlygy emele getiren sebäp anyklandy;
- ýalňyşyň ýa-da beýleki ýüze çykan meseläniň gutarnykly seljermesi geçirildi;
- düzedişiň üstünde iş başlanlygy barada çözgüt kabul edilen;
- ýalňyş düzedildi;
- düzediş programma önümiň düzümine girdi;
- ýalňyş düzedilenligi tassyklanyldy.

21.2. Programma önümlerini ýola goýmak

Ýola goýma – bu ýalňyşlyklary çäklendirmek (lokallaşdyrmak) we ýok etmek prosesi. Testirme we ýola goýma prosesleri 18.1-nji suratdaky ýaly shematiki görkezmek bolar.

Programmany ýola goýmak prosesi örän döredijilikli prosedir. Emma ýola goýmaklygyň esasy ideýasyna (maglumatlaryň seljerilişine esaslanan) indiki algoritmiň görnüşini bermek bolar:

1. Eýýäm elýeterli bolan we jemleýji maglumatlary öwrenmek.

2. Şeýle jemleýji maglumatlaryň alnyşsyny düşündirýän käbir gipotezany kesgitlemek.

3. Täze giriş maglumatlary taýýarlamak we gipotezany subut etmäge ýa-da onuň ýalandygyny anyklamaga mümkinçilik berýän eksperimenti geçirmeli.

Ýola goýmaklygyň iki sany ýönekeý ýagdaýyna üns bereliň. Olaryň seljerilişi ädimleýin ýola goýmaklygyň üsti bilen aňsat geçirilip bilner:

- eger programma öz işini «Ýadyň dampyny» (core-faýly) döretmek bilen tamamlayan bolsa, onda islendik ýola goýujy bu dampa proseduralary çagyrmaklygyň dinamiki akymyny dikeltmäge we mümkin ýalňyş ýerini çäklendirmäge mümkinçilik berer;
- eger programma korrekt däl habary çykarýan bolsa, onda saklaýyş nokady hökmünde habarlary çykaryjy funksiýanyň çagyrylyşyny alyp, ýola goýujyda dinamiki akymy almak bolar. Akym pikirlenmek üçin ýeterlik maglumatlary berip biler.

Kompýuter adalgalarynda programmada ýüze çykan ýalňyşa bag diýilýär (inl. dilinden terjime edeniňde – tomzak). Bu adalganyň emele gelşini 1947-nji ýylda «Mark II» kompýuteriniň relesinden çykarylýp alnan tomzajyk bilen baglaşdyrýarlar.



Programmany işe girizmek

Programmany işe girizmek onuň niýetlenilişine bagly. Programma önümi bellibir buýrujy üçin ýa-da giňden ulanylyş üçin bolup biler. Ulanyjy üçin programmany eltmek usulynyň üç görnüşi bar:

- indiividual eltmeklik (düzgün bolşy ýaly, bellibir buýrujy üçin);
- gutujykda eltmeklik;
- internetiň üsti bilen eltmeklik.

Soňky görnüşde eltmekligiň ýene-de birnäçe görnüşi bar:

- programma bilen işleýiş Internetiň üsti bilen amala aşyrylýar. Düzün bolşy ýaly internet-brauzeriň gurşawyndan ula-

nyjlylar programmanyň ýerine ýetiriji koduny (ýa-da onuň müşderiler üçin bölegini) tor boýunça alyp, programmanyň iň soňky görnüşini bilen işläp başlap bilýärler. Şeýle goşundylaryň köpüsi bilen ulanyjylar mugt işleýärler. Emma kompaniýalaryň birnäçesi käbir programma önümi bilen işleýiş serwisleri döwürleýin töleg esasynda satmaklyk ideýasyny ösdürýärler;

- internet torundan programmany almak we kompýutere ornaşdyrmak mümkiçiligi programmany gutujykda eltmeklige meňzeş. Internet programma üpjünçiligini ýaýratmak we ulanyja eltmek meselesini örän ýönekeý çözüýär. Gutujykdaky önümleriň (1985–1995) ýerine «Internet önümleri» geldiler. Internetde programma önümleriň iki topary bar:

- erkin ýaýradylýan programma önümleri;
- täjirçilik programma önümleri. Bu önümlere ilki wagtlaýyn mugt registrasion açary (adatça, bir aýlyk) alyp bolýar, soňra programmanyň artykmaçlyklaryny we kemçiliklerini ýüze çykaryp bu önümi satyn almaklyk barada çözügüt kabul etmek bolar. Eger programma önümi ulanyjy üçin gabat gelýän bolsa, onda programmanyň bahasyny töläp hemişelik registrasion açary alyp bilýär.

Programmanyň instalýasiýasy ýaly işiň soňky döwürde köp ýagdaýlarda (mysal üçin, programma bilen Internetiň üsti bilen işlemek) ulanylmaýandygyny bellemek gerek. Emma instalýasiýany taýýarlamak prosesinde talabalaýyk düzgünleriň käbirlerini getireliň (http://www.softshape.com.swrus/faq_swrus.html):

- programmany bir özbaşdak açylýan faýla ýygnaýan instalýatory ulansak-da, programmany zip-arhiwa gysyp ýaýratmak maslahat berilýär. Olar faýly has gowy gysýan bolsalar-da, beýleki arhiwleýji programmalary ulanmak maslahat berilmeýär. Sebäbi ulanyjyda bu arhiwleýji programma ýok bolmagy ahmal;
- arhiwa hemişe programma barada maglumaty saklaýan RE-ADME tekst faýly goşmaly. Bu faýl ulanyja instalýasiýadan öň elýeterli bolmaly;

- instalýator programmany kompýuterden awtomatiki ýok edýän mümkinçiligi saklaýandyga göz ýetirmek gerek. Deinstalýatorsyz programma ulanyjylara köp goşmaça oňaýsyzlyklary döredýär.



23. Programma önüminiň ekspluatasiýasy we alyp barlyşy

Programmany alyp barmak diýmek, bu programma önüminiň ygtybarlylygyny ýokarlandyrmak boýunça ähli ýerine ýetirilmeli işleriň jemine we kämilleşdirilen görnüşleriň işlenilip taýýarlanylmagyna aýdylýar. Alyp barmaklygyň esasy düzgüni bolup, Laslo Bilediniň kanuny çykyş edýär. Bu kanun boýunça ulanylýan programma önümi, onuň ykdysady bähbidini goldamak üçin, hemişe üýtgedilip durulýar.

Programma önümini alyp barmak tapgyrynda çözülýän esasy dört meseläni sanap geçeliň:

- *adaptasiýa*. Ol funksiýalaryň modifikasiýalaryndan ybarat;
- *kämilleşdirmek*. Ol täze funksiýalary goşmakdan ybarat;
- *korreksiýa* ýa-da ýalňyşlary düzetmek;
- geljekde emele gelip biljek meseleleriň *öňüni almak*.

Meseleleriň bu kategoriýalary şertlidir we mesele bir kategoriýadan beýleki kategoriýa geçirilip bilner.

Bu meseleleri çözmek üçin alyp barmaklygyň aşakda getirilen görnüşleriniň has gabat gelýänini saýlamak gerek:

- *ähmiýetli däl (ýerli) üýtgetmeler*;
- *koduň restrukturirlenmegi* – programmanyň uly däl bölegini gaýtadan işlemeklik. Şunlukda, bu bölegiň galan programma bilen interfeýsi üýtgemän galýar;
- *reinziniring* – bar bolan programma önümini täzedan düzmek. Adatça, reinziniring programmanyň koduny programmirlemegiň häzirki zaman usulyýetlerine (mysal üçin, obýekte-gönükdirilen) laýyk getirmekligi üçin ulanylýar;
- *täzedan programmirlemek*.

Bu alyp barmaklygyň görnüşleri programma goşulmak derejesi boýunça bölünýärler. Alyp barmaklygyň görnüşine iki faktor täsir edýär:

- *çözülyän meseläniň synpy.* Düzgün bolşy ýaly, programmanyň koreksiýasy programma sähel derejede goşulmaklyga getirýär. Kämilleşdirmek bilen adaptasiýa bolsa – restrukturirlenmegi we reinziniringidir;
- *alyp barmaklygyň bahasy.* Alyp barmaklygyň ýokary bahasynda restrukturirlenmek bilen reinziniringe, pes bahasynda bolsa ähmiýetli däl üýtgeşmelere gönükmek maslahat berilýär.

Tejribäniň görkezişi ýaly, programmany alyp barmaklyk bilen tejribe programma düzüjiler meşhullansalar ol has netijeli bolup, önümiň konseptual bitewiliginiň bozulmagyna getirmez.

Programmany alyp barmaklygyň ýokary hilini üpjün etmäge mümkinçilik berýän usullaryň birnäçesi bar:

- *üýtgemeleriň baglaşmak çägin belläp almak.* Programmanyň üýtgedilýän ýerleriniň çäklerini anyk bilmek gerek;
- *üýtgemeleriň özbaşdak resminamalaşdyrylmagy;*
- *üýtgemelere passiw we aktiw garşylyk.* Programmany üýtgetmekden öň programma düzüji programmany üýtgetmek barada we üýtgedilýän koduň böleklerini saklaýan galan haty işläp düzüjilere ugratmaly. Programma düzüji üýtgetmeleri diňe indiki ýagdaýlarda:
 - eger işdeşleriň hiç biri garşy çykmasa (passiw garşylyk ýagdaýynda);
 - eger işdeşleriň ählisi makullasalar (aktiw garşylyk ýagdaýynda) girizip bilýär.

Alyp barma prosesini kompilýatorlar ýaly programmalaryň mysalynda getireliň. Olary alyp barmaklyga häsiýetli bolan aýratynlyklaryna eýe bolany üçin, bu aýratynlyklary has umumylaryndan başlap has spesifikilerine çenli tertipde sanap geçeliň:

- birnäçe apparat platformalarynda işlemeklik;
- ulanyjylaryň örän köp bolmaklygy;
- jikme-jik testirmekligiň zerurlygy;
- könelişen algoritmeleriň bolmagy;
- programmirleme dillerine standartlaryň bolmaklygy.

Alyp barmak meseleleriniň esasy synplarynyň çeşmeleri bolup, aşakdakylar çykyş edýärler:

- ýalňyşlary tapýan kompilýatoryň ulanyjylary;

- özara baglanyşykly komponentleri işläp düzüjiler;
- kompilýatory alyp barýan inženerler (mysal üçin, kompilýatoryň netijeliligini artdyrmaklygy hödürlep);
- täze algoritmleri we optimizasiýalary oýlap tapýan alymlar.

Indi alyp barmaklygyň meseleleriniň esasy dört synplaryň kompilýatora degişlilikde, çözülişine seredeliň:

- kompilýatorlaryň adaptasiýasy ulgamyň üýtgän iş şertlerine uýgunlaşmaklygyny talap edýär. Üýtgemekligiň ähli sebäpleri üç sany esasy sebäplere jemlenýär:
 - *arhitektura degişli üýtgemeler*. Mysal üçin, adaptasiýanyň esasy görnüşi bolup kompilýatory beýleki arhitektura geçirmek çykyş edýär;
 - *operasion ulgamynda üýtgemeler*. Mysal üçin, operasion ulgamy köp akymly ýerine ýetirilişi käbir wersiýadan başlap goldap başlap bilýär;
 - *özara baglanyşykly komponentlerde üýtgemeler*. Mysal üçin, programmanyň tekstine ähmiýetli bolmadyk düzedişleri girizmäge we täzeden kompilirlmeksiz ýerine ýetirilişi dowam etmäge mümkinçilik berýän aragatnaşyklaryň inkremental redaktorynyň işlenilip düzülmegi;
- kompilýatorlaryň kämilleşdirilmegi buýrujynyň islegine laýyklykda olara üýtgemeleri girizmekligi göz önünde tutýar. Kämilleşdirmekligiň birnäçe esasy ugurlaryny belläliň:
 - *netijeliligini ýokarlandyrmak*. Mysal üçin, semantiki seljermäniň algoritmleriniň optimizasiýasy;
 - *ygtybarlylygy ýokarlandyrmak*;
 - *diliň standartlaryna laýyk gelmekligiň zerurlygy*. Standart üýtgeýän ýagdaýynda ýa-da standarta kommentarileriň döredilmegi ýagdaýynda kompilýator bu üýtgemeleri hasaba almalydyr;
 - *diliň meşhur dialektlerine laýyk gelmekliginiň zerurlygy*. Käwagt standarta doly gabat gelýän kompilýator beýleki meşhur kompilýator bilen işlenmeklige goýberilýän konstruksiýalarynyň işlenilmegine ýol bermeli. Adatça, bu ykdysady bähbitligi bilen düşündirilýär;

- eger kämilleşdirmek bilen adaptasiýa programmirleme diline görkezme berilmän, köp kompilyatorlar bilen seredilip bilinýän bolsa, ýalňyşlaryň korreksiýasy köp derejede bellibir dilli kompilyator üçin gönükdirilendir.

Köplenç, programma önümini alyp barmaklyk bilen ony işläp düzen kompaniýa meşgullanýar. Emma önümiň alnyp barylýşynyň beýleki kompaniýa berilýän ýagdaýlary hem bolup biler.

Ekspluatasiýany tamamlamak

Programma önüminiň ekspluatasiýasyny tamamlamak prosesi çylşyrymly däl, emma ol dykgatly guramaçylykly taýýarlygy talap edýär. Adatça, proses ulanyjylaryň programma önümini alyp barmaklygy tamamladygy barada habar bermekliginden başlaýar. Şol pursatdan başlap ulanyjylar programma önümini ol moral taýdan könelýänçä ulanmaklygy dowam edip bilerler. Köplenç, programmany alyp barmaklygyň dört meselesiniň birini-de çözüp bilmeklik önümi kabul etmezlige eltýär.

IV. PROGRAMMA ÜPJÜNÇİLİĞİNİ DÖRETMEKLİĞİN TEHNOLOGIKI ÇEMELEŞMELERİ



Programmirleme tehnologiýalarynyň çemeleşmeleriniň klassifikasiýasy

Tehnologiki çemeleşmeleriň esasy toparlaryny belläliň we olaryň her biri üçin hem çemeleşmeleri görkezeliň.

Gowşak formalizasiýaly çemeleşmeler

Bu çemeleşmeler anyk tehnologiýalary ulanmaýarlar we olary diňe, adaty, görkeziji prototipini döretmek bilen tamamlanýan kiçi taslamalar üçin ulanmak bolar. Gowşak formalizasiýaly çemeleşmeler irki tehnologiki çemeleşmeler, mysal üçin «kodlamak we düzetmek» degişlidir.

Berk (klassiki, berk, öňünden belli) çemeleşmeler

Çemeleşmeleriň bu toparyny ulgama anyk talaplary bolan orta, uly gerimli we örän uly taslamalar üçin ulanmak maslahat berilýär. Şeýle taslamalara edilýän esasy talaplaryň biri-de öňünden görüjilik. Bu topara indiki çemeleşmeler girýär:

- kaskad görnüşli tehnologiki çemeleşmeler:
 - klassiki kaskad çemeleşmesi;
 - kaskad-gaýdyş çemeleşme;
 - kaskad-iterasion çemeleşme;
 - biri-biriniň üstünü ýapýan prosesli kaskad çemeleşme;
 - podprosesli kaskad çemeleşme;
 - spiral modeli;
- karkad görnüşli tehnologiki çemeleşmeler:
 - rasional unifikirlenen proses;
 - Microsoft Solution Framework prosesleriň modeli.

Formal çemeleşmeler

Formal çemeleşmeler programma üpjünçiligini döretmek prosesine aýratyn formal talaplary göz önünde tutýarlar. Mysal üçin, gene-

tiki çemeleşmeler üçin programmanyň döreyşi we ony döretmekligiň tertibi bilen bagly bolan formallyklary talap edýärler. Bu topara indiki çemeleşmeler girýär:

- genetiki çemeleşme:
 - sintezirleýän programmirlеме;
 - ýygnama (giňeldilýän) programmirlemesi;
 - anyklaşdyrýan programmirlеме;
- formal özgertmeler esasyndaky çemeleşmeler:
 - arassalanan sehiň tehnologiýasy;
 - formal genetiki çemeleşmeler.

Çeýe (adaptiw, ýeňil) çemeleşmeler

Bu toparyň çemeleşmeleri ulgama talaplaryň üýtgeýän ýa-da anyk belli bolmadyk ýagdaýlarynda, uly däl ýa-da ortaça taslamalar üçin ulanmak maslahat berilýär. Programmany düzüjileriň topary jogapkärçilikli we kwalifisirlenen bolmaly, buýrujylar bolsa programmany işläp düzmek prosesine gatnaşmaga razy bolmaly. Bu topara indiki çemeleşmeler girýär:

- irki tiz işläp taýýarlamaklygyň tehnologiiki çemeleşmeleri:
 - ewolyusion prototipirlеме;
 - iteratiw işläp taýýarlaýys;
 - döwürleýin işläp taýýarlaýys;
- adaptiw çemeleşmeler:
 - ekstremal programmirlеме;
 - adaptiw işläp taýýarlaýys;
 - Crystal tehnologiiki çemeleşmeleriň maşgalasy;
- barlag programmirlемегин çemeleşmeleri:
 - kompýuter darwinizmi;
 - fragmentar programmirlеме.



25. Kaskad görnüşli tehnologiiki çemeleşmeler

Kaskad görnüşli tehnologiiki çemeleşmeler, adatça, kaskad görnüşde aňladylýan prosesleriň ýerine ýetirilişiniň käbir yzygiderligini berýärler.

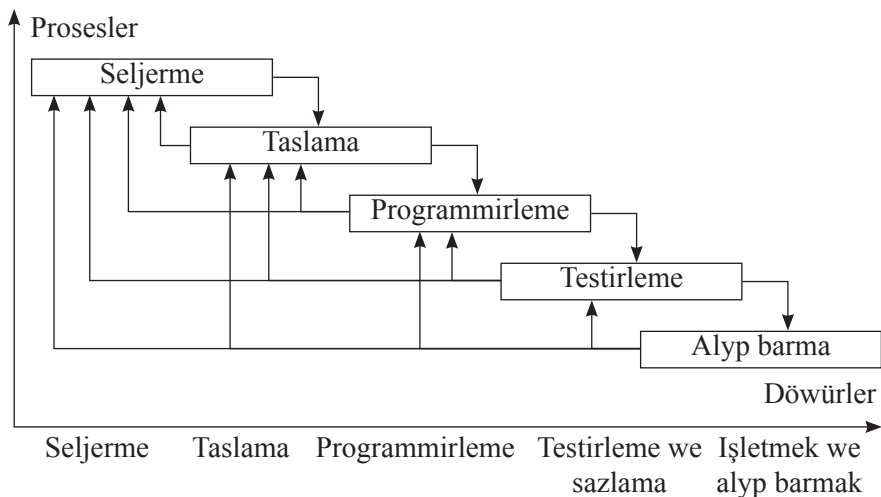
Klassiki kaskad çemeleşmesi

Klassiki kaskad çemeleşmesi (pure waterfall) programmanyň ýaşayyş siklini alyp barmaklygyň tehnologiكي çemeleşmeleriniň iň irkisi hasaplanýar. Bu çemeleşmä beýleki çemeleşmeleriň köpüsi üçin başlangyç nokady hökmünde garamak hem bolar. Bu çemeleşme 1970-nji ýyldan, 1985-nji ýyllar aralygynda emele geldi. «Arassa» kaskad çemeleşmäniň spesifikasiýasy boýunça indiki prosese geçmeklik diňe işlenilip duran prosesini işi doly tamamlanandan soň amala aşyrylýar. Eýýäm geçilen proseslere yza gaýtmanyk göz önünde tutulmadyk.

Bu çemeleşme taslama edilyän talaplar başdan doly we takyk kesgitlenen ýagdaýlarda ulanmak maslahat berilýär. (Mysal üçin, hasaplaýyş häsiýetli meselelerde). Şeýle tehnologiكي çemeleşme esasynda işleriň alnyp barlyşyny meýilnamalaşdyrmak we býujeti emele getirmek ýaly işleri alyp barmak aňsat düşýär.

Kaskad-gaýdyş çemeleşmesi

Klassiki kaskad çemeleşmäniň esasy kemçiligi – bu çyýeligiň ýoklugydyr. Hut şu kemçiligi ýeňip geçmek üçin kaskad-gaýdyş çemeleşmesi ulanylýar. Bu çemeleşmede öňki döwürlere gaýdyp gelmäge we ön kabul edilen çözgütlere täzeden seredip, anyklamaga rugsat berilýär (25.1-nji surat). *Kaskad-gaýdyş çemeleşmesi* programma üpjünçiligini işläp düzmekligiň iterasion häsiýetini görkezýär. Bu



25.1-nji surat. Kaskad-gaýdyş görnüşli tehnologiكي çemeleşme

çemeleşme programma üpjünçiligini döretmekligiň hakyky prosesini köp derejede (şol sanda hem netijäni almak bilen esli gijä galmaklyk) görkezýär. Yza gaýdyp düzedişleri girizmek hem gijä galmaklyga örän täsir edýärler.

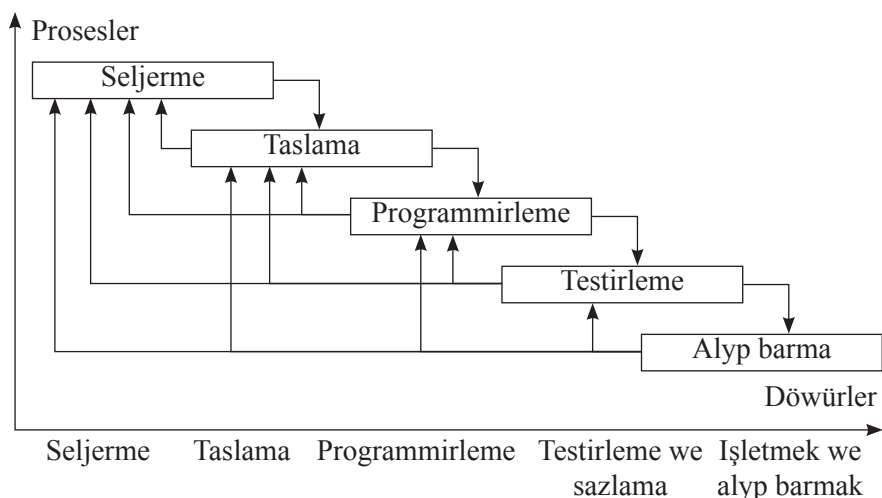
Aşakda taslamanyň ýaşaýyş siklini alyp barmakda kaskad-gaýdys tehnologiýa çemeleşmäniň shemasy getirilen.

Kaskad-iterasion çemeleşme

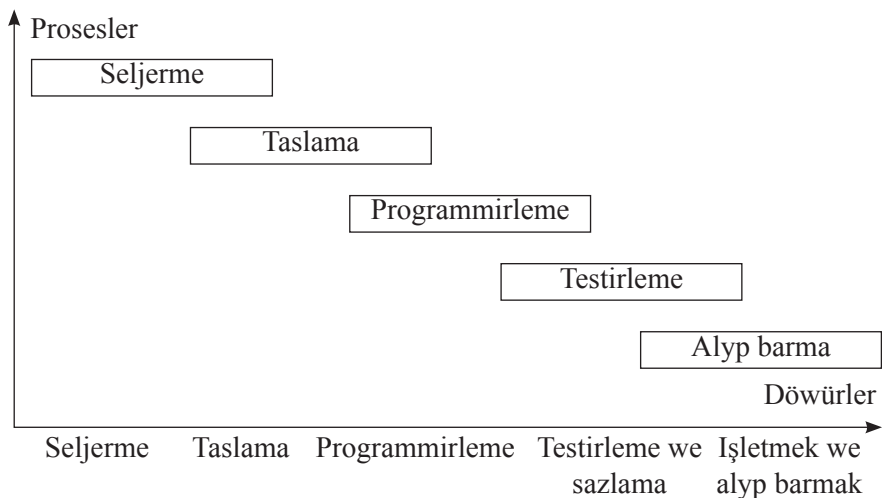
Kaskad-iterasion çemeleşmesi her prosesniň yzygider iterasiýasyny gerekli netije alynýança amala aşyrmaklygy göz önünde tutýar (25.2-nji surat). Her bir iterasiýa gutarnykly proses hasap edilmeýär we onuň netijesi bolup käbir aralyk jem çykyş eder. Alnan netije aralyk bolmaklygy ahmal, emma ol garaşylýan funksionallýgy amala aşyýar.

Biri-biriniň üstüni ýapýan prosesli kaskad çemeleşme

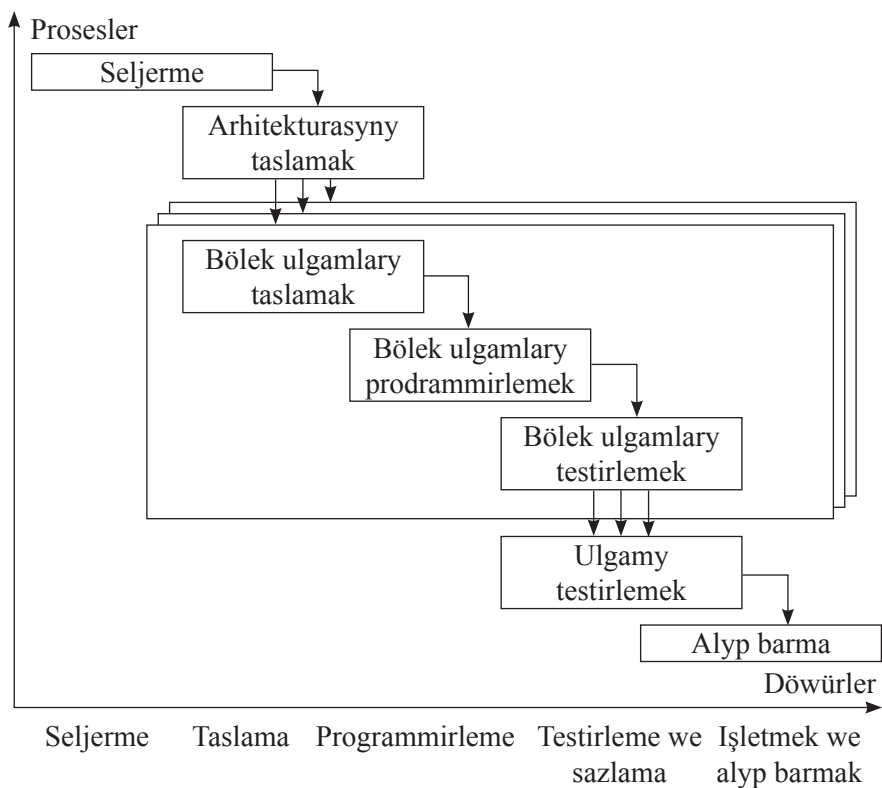
Klassiki kaskad çemeleşmesi her prosesi aýratyn buýruk bilen ýerine ýetirmäge mümkinçilik berýär. *Biri-biriniň üstüni ýapýan prosesli kaskad çemeleşmesi (waterfall with overlapping)* bolsa iberilýän maglumatlary bellibir derejä çenli gysgaltmaga mümkinçilik berýän ýörite buýruklaryň bolmagyny göz önünde tutýar. Indiki proses ondan öňküler gutarmanka başlaýar (25.3-nji surat). Ondan hem başga birnäçe prosesler parallel hem ýerine ýetirilip bilinýärler.



25.2-nji surat. Kaskad-iterasion görnüşli tehnologiýa çemeleşme



25.3-nji surat. Biri-biriniň üstüni ýapýan prosesli kaskad çemeleşme



25.4-nji surat. Podprosesli kaskad çemeleşme

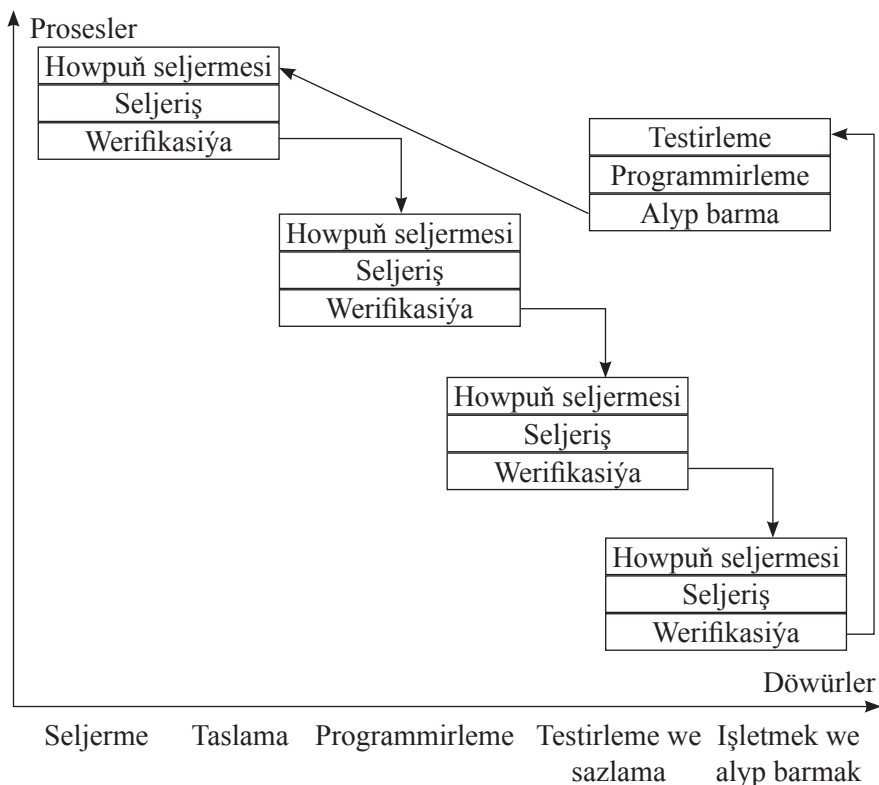
Podprosesli kaskad çemeleşme

Podprosesli kaskad çemeleşme (waterfall with subprocesses) biri-biriniň üstüni ýapýan prosesli kaskad çemeleşmä örän ýakyn. Arhitektura nukdaýnazary boýunça taslama köplenç, aýratynlykda işlenilip düzülýän, bölek taslamalara bölünip bilýär. Hut şu aýratynlyk podprocessli kaskad çemeleşme üçin häsiýetlidir (25.4-nji surat).

Bu çemeleşmede bölek ulgamlary bir umumy ulgama birikdirmek üçin bölek ulgamlary testirmekligiň goşmaça fazasy zerur. Şeýle hem taslamany bölek taslamalara dogry bölmeklige aýratyn üns bermeli, sebäbi bu bölünme bölek ulgamlaryň arasyndaky ähli mümkin boljak aragatnaşyklary hasaba almalydyr.

Spiral modeli

Spiral modeli (spiral model) Barri Boem tarapyndan 80-nji ýyllaryň ortalaryna işläp düzmekligiň mümkin bolan howpuny pe-



25.5-nji surat. Spiral modeli

seltmek maksady bilen hödürlendi. Spiral modeli prototip – döredilýän programma önüminiň bölekleyin funksionallygyny amala aşyran programmany, düşüňjesini ulanýar. Prototipleri döretmek prosesi spiralyň birnäçe aýlawy bilen amala aşyrylýar. Her bir aýlaw «Howpuň seljermesi», «Käbir proses» we «Werifikasiýa» ýaly proseslerden ybarat (25.5-nji surat). Her bir prosese ýüzlenmekligi «Howpuň seljermesi» amala aşyryar. Özi hem eger-de taslamany ýerine ýetirmekligiň möhletini geçirmek we taslamanyň bahasyny geçirmek çenden aşa bolsa, onda işläp taýýarlaýyş prosesi tamamlanýar. Bu geljekde pul çykadjylary çenden aşa sarp etmekligiň önüni almaklyga mümkinçilik berýär.

Spiral modeliniň aýratynlygy – iterasiýalar bilen işlemekdedir. Özi hem her bir indiki iterasion prototip has uly funksional mümkinçilige eýe bolar.



26. Karkas görnüşli tehnologiki çemeleşmeler

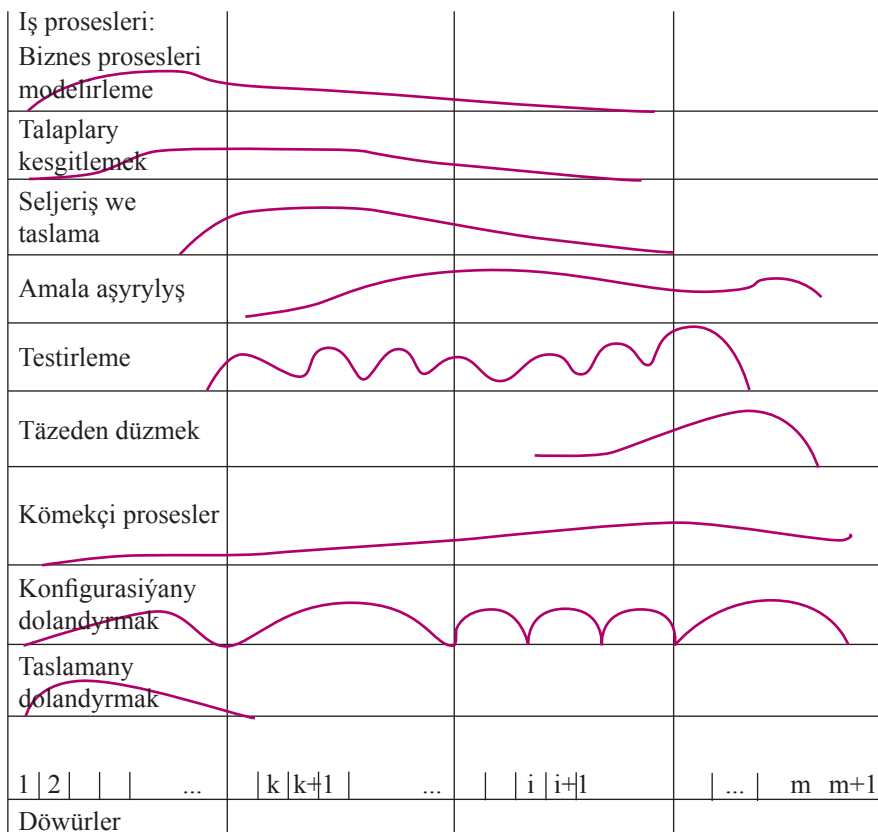
Karkas çemeleşmeler prosesler üçin karkasy emele getirýärler we olaryň köp sanyny öz içine alýarlar.

26.1. Rasional unifikirlenen proses

Rasional unifikirlenen prosesi (rational unified process) kaskad görnüşli tehnologiki çemeleşmeleriň in gowdy taraplaryny öz içine aldy. Onda dört sany esasy fazalary tapawutlandyrýarlar. Hut şu fazalary geçmekligiň dowamynda yzygider iterasiýalardan ybarat bolan prosesler hem funksionirlenýärler (26.1-nji surat).

Bu çemeleşmäniň esasy aýratynlyklary:

- iteratiwlik we oňa mahsus bolan çyýelik;
- hili barlamak, mümkin boldugyça howplary ýüze çykarmak we ýok etmek;
- kagyz resminamalaryň ýerine modeller bilen işlemeklik makullanylýar;



26.1-nji surat. Rasional unifikirlenen proses

- arhitekturanyň kesgitlenmegine aýratyn üns berilýär;
- konfigurirlemek, ýola goýmak we masştabirlemek mümkinçiligi.

26.2. Microsoft Solution Framework processorlarynyň modeli

Microsoft Solution Framework processorlarynyň modeli (MSF) (<http://www.microsoft.com/msf/>) – programma önümleri ýaýradylanda, şeýle hem biznesiň meselelerini çözmekde, tehnologiýalary has netijeli ulanmaklygy üpjün edýän modelleriň, düzgünleriň we spesifikasiýalaryň toplumyna esaslanýan tehnologiýa çemeleşmedir.

MSF modeli Microsoft kompaniýasynda programma goşundylary döretmekde ulanylýan çemeleşmäniň esasynda döredi. Özüniň ösüşiniň netijesinde bu çemeleşme beýleki belli bolan prosesleriň modellerini has netijeli düzgünleriniň birnäçesini birleşdirdi we onuň netijesinde islendik görnüşli taslamalaryň üstünde işlemek üçin bir ýeke-täk bazasyny emele getirdi.

MSF prosesleriň modeliniň üç aýratynlyklary bolup aşakdakylar çykyş edýärler:

- *fazalara we pursatlara esaslanan çemeleşme*. Pursatlar taslamanyň gidişiniň monitoringi we meýilnamalaşdyrylyşy üçin daýanç nokatlar hökmünde ulanylýarlar. Esasy pursatlar – şol ýa-da beýleki fazada alnan netijeler taslama toparynyň agzalarynyň biri-biri bilen we buýrujynyň islegi bilen sinhronizirlenende taslamanyň ýaşayyş sikliniň pursatlary. Bu pursatda buýrujy dahylly taraplar we taslama topary bilen alnan progresiň formal seljermesi geçirilýär. Onuň üstünlikli geçmegi bolsa taslamanyň üstünde işi dowam etmekligi aňladýar;
- *iterativ çemeleşme*. Programma kody, resminamalar, dizaýn, meýilnamalar we beýleki iş gurallary düzgün bolşy ýaly iterativ usullar bilen döredilýärler. MSF çözgüdi işlemekligi, onuň bazalaýyn funksionallygyny gurmak, testirmek we ornaşdyrmakdan başlamaklygy maslahat berýär. Ondan soň çözgüde täze-täze mümkinçilikler goşulýarlar;
- *çözgütleri tapmaklyga we ornaşdyrmaklyga integrirlenen çemeleşme*. MSF prosesleriň modeli çözgüdi döretmegiň ähli ýaşayyş siklini, şol sanda, onuň ornaşdyrylmagy we çözgüdiň jogap berip başlaýan pursadyna çenli öz içine alýar.

Indi bolsa MSF prosesleriň modeliniň fazalaryny we pursatlaryny sanap geçeliň:

- *konsepsiýany işläp düzmek (envisioning phase)* fazasynda taslamanyň üstünliliginiň fundamental esaslarynyň biriniň düýbi tutulýar. Ol esas – bir umumy görüşi emele getirmekligiň esasynda taslama toparyny döretmek we jebislemek. Taslama topary buýrujy üçin näme döretjegini anyk göz önüne getirmeli we buýrujyny hem taslama toparyny hem maksimal motiwirmek üçin öz maksadyny kesgitlemeli;

- *meýilnamalaşdyryş (planning)* fazasynda taslamanyň meýilnamasyny düzmek boýunça esasy işler amala aşyrylýarlar. Ol taslama topary bilen funksional spesifikasiýany taýýarlamak, dizaýnlary işläp taýýarlamak, iş meýilnamalaryny taýýarlamak, taslama çykadjylary bahalandyrmak we taslamanyň dürli bölümlerini işläp taýýarlamak üçin möhletleri bellemek ýaly işleri öz içine alýar;
- *işläp düzmek (developing)* fazasynda taslama topary çözgüt komponenti (resminamalary hem programma koduny hem öz içine alýar) döretmeklikde fokusirlenýärler. Emma bu işiň käbir bölegi eger testirmek prosesinde beýle zerurlyk ýüze çykan bolsa, stabilizasiýa fazasynda dowam edip biliner. Bu faza infrastrukturany işläp düzmekligi hem öz içine alýar;
- *stabilizirleme (stabilizing)* fazasynyň dowamynda işlenilip düzülen çözgüdiň testirlenmegi amala aşyrylýar. Taslama topary ýalňyşlyklaryň ilkinjiligi we olaryň aradan aýrylmany, şeýle hem programmany çykarylyşa taýýarlamaklygy bilen meşgullanýar;
- *ornaşdyrmak (deploying)* fazasynyň dowamynda taslama topary tehnologiýalary we çözgüdiň komponentlerini ornaşdyrýar, ornaşdyrylan çözgüdi stabilizirleýär, işi goldamak we alyp barmak prosesiniň personalyna geçirýär we buýrujy tarapyndan taslamanyň netijeleriniň jemleýji goldawuny alýar. Ornaşdyrmak tamamlanandan soň taslama topary geçirilen işiň we buýrujynyň kanagatlanarlyk seljermesini geçirýär.



27. Genetiki hem-de tehnologiiki çemeleşmeler

Bu toparyň ady I.W.Pottosin bilen ilkinji gezek girizildi we «Genetiki» adalgasy programmanyň emele gelşi we ony döretmek dersi bilen baglanyşdyrylýar.

27.1. Sintezirleýän programmirleme

Sintezirleýän programmirleme programmanyň sintezini onuň spesifikasiýasy boýunça amala aşyrmaklygy talap edýär. Algoritm dilinde ýazylan we kompýuterde ýerine ýetirmek üçin

niýetlenen programmadan tapawutlylykda spesifikasiýalar dilinde ýazylan resminama indiki amala aşyrylyş üçin diňe bazis bolup durýar. Bu amala aşyrylyşy almak üçin indiki esasy meseleleri çözmek zerur:

- spesifikasiýalar diliniň kömegi bilen aňladyp bolmaýan ýerine ýetiriji kody almak üçin zerur bolan detallary kesgitlemeli;
- amala aşyrylyş dilini we amala aşyrylyşynyň apparat-programma platformasyny saýlamaly;
- spesifikasiýalar diliniň düşünjelerini amala aşyrylyş diline we apparat-programma platformasyna aňladylyşyny bellige almak;
- aňladylyşyň (spesifikasiýadan amala aşyrylyş dilindäki ýerine ýetiriji programma) transformasiýasyny amala aşyrmak;
- ýerine ýetirilýän programmany testirmek we ýola goýmak.

Eger-de spesifikasiýa meseläni çözmekligiň usulynyň doly beýanyny saklamaýan bolsa (mysal üçin, kompilyator üçin giriş tekstiň semantiki taýdan dogrulygynyň usuly beýan edilmedik bolsa), onda programmanyň awtomatiki generirlenmegiň mümkin daldigini hem bellemek gerek.

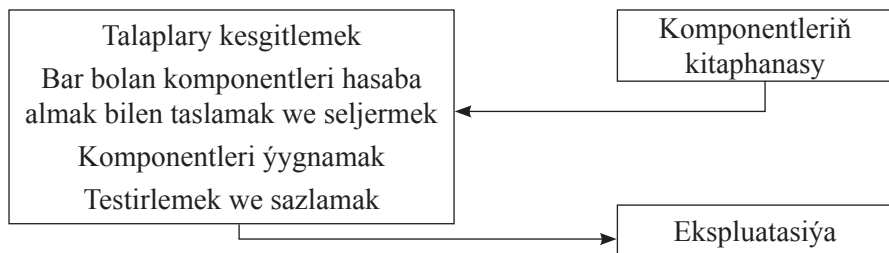
Spesifikasiýalar boýunça programmalaryň awtomatiki generirlenmegi dilleriň köpüsi üçin ýerine ýetirilip bilner. Olaryň arasynda SDL, ASN.1, LOTOS, Estelle we UML aýratyn bellemek bolar.

27.2. Ýygnama (giňeldilýän) programmirleme

Ýygnama programmirleme, programmanyň eýýäm belli bolan bölekleri (fragmentleriň) täzedan ulanmak bilen ýygnalýandygyny göz önünde tutýar (27.1-nji surat).

Ýygnama elde ýa-da ýygnamaklygyň käbir dilinde amala aşyrylyp bilner ýa-da meseläniň spesifikasiýasyndan ýarym awtomatiki görnüşde alnyp bilner. Aşakda ýygnama programmirlemegiň tehnikasyny döretmekligiň esasy ugurlary getirilen:

- kabul edilen modullylyk prinsipine laýyk gelýän *programmirleme stilini* işläp çykarmak;
- *modulara interfeýsleriň netijeliligini* ýokarlandyrmak; modullylygyň apparat goldawynyň wajyplygy;



27.1-nji surat. Ýygnama programmirleme

- *programma modullaryň uly bazasyny* alyp barmak, modullaryň identifikasiýasy meselesini çözmek we interfeýsiň beýany boýunça ulanyp bolujylyk mümkinçiligini barlamak. Modullar, programmany emele getirýän «Programma kerpiçleri» bolup durmaly.

Ýygnama programmirleme başlangyç hem binar hem kodlarynyň täzeden ulanmak usuly bilen ýakyn baglanyşyklydyr. Ýygnama programmirlemegiň birnäçe dürli tehnologiýa çemeleşmelerini bel-
leýärler:

- *modullaýyn ýygnama programmirleme.* Bu çemeleşme taryhy taýdan ilkinji bolup, düzümlýän imperatiw programmirleme usulyýetiniň funksiýalaryna we proseduralaryna esaslandy;
- *obýekte gönükdirilen ýygnama programmirleme.* Çemeleşme obýekte gönükdirilen programmirlemegiň usulyýetine esaslanan we synplaryň kitaphanasyny başlangyç kod görnüşinde ýaýratmaklygy ýa-da synlary dinamiki kitaphana ýygna-
maklygy talap edýär;
- *komponent ýygnama programmirleme.* Çemeleşmäniň esasy taglymatlary – synlary binar görnüşinde ýaýratmak we synpyň usullaryna elýeterligi berk kesgitlenen interfeýsleriň üsti bilen amala aşyrmak. Bu kompilytorlaryň ylalaşylyk meselesini aradan aýyrmaklyga mümkinçilik berýär we synplaryň görnüşleriniň çalşyrmasy, olary ulanýan goşundy programmalary täzeden kompilirlemeksiz üpjün edýär. Komponent ýygnama programmirlemegi goldaýan anyk tehnologiýa çemeleşmelere COM (DCOM, OM+), CORBS, .NET degişlidirler;

- aspekte gönükdirilen ýygnama programmirleme. Bu ýagdaýda komponentiň konsepsiýasy aspektiň konsepsiýasy – netijeliligi boýunça kritiki proseduralaryň amala aşyryşynyň bir görnüşi bilen doldurylýar. Aspekte gönükdirilen ýygnama programmirleme, amala aşyrylyşyň dürli görnüşlerini inkapsulirleýän, köp aspektli komponentlerden doly funksional goşundy programmalary ýygnamaklygyndan ybarat.

27.3. Anyklaýjy programmirleme

Anyklaýjy programmirleme hususy, ýörite programmalaryň uniwersal programmalaryndan alynmagyny talap edýär.

Anyklaýjy programmirlemegiň iň belli bolan tehnologiýasy – taslamaklygyň patternlerini ulanmak bilen çemeleşme. *Taslamaklygyň patterni (şablony) (design pattern)* – anyk bir kontekstde taslamaklygynyň umumy meselesini çözmek üçin adaptirlenen obýektleriň we synplaryň aragatnaşygynyň beýany. Patternler bilen tanyş bolan taslaýjy, olary täze meseläni çözmek üçin göni ulanyp biler. Taslamaklygyň patternleri şowly taslaýjy we arhitektura çözügütleri täzedan ulanmaklygyny aňsatlaşdyrýarlar.

Pattern dört sany esasy elementden ybarat:

- taslamaklygyň meselesini anyk beýan edýän ady;
- anyklamak üçin patterni ulanmaklygyň wagt beýanynyň meseleleri;
- dizaýnyň elementleriniň abstrakt beýanynyň we olaryň arasyndaky gatnaşyklaryň *çözügütleri*;
- patterni ulanmaklygyň *netijeleri*.

Patternlere goşmaça *karkaslar (framework)* hem ulanylýar. Karkas – programmalaryň bellibir synplary üçin gaýtadan ulanylýan dizaýny düzýän özara baglanyşykly synplaryň toplumlary. Karkas goşundy programmanyň kesgitlenen arhitekturasyny talap edýär we onda taslama çägi üçin umumy taslama çözügütleri ýygnaýan. Mysal üçin, kompilyatorlary işläp düzmek üçin karkaslar hem bar.



28. Formal özgertmeler esasyndaky çemeleşmeler

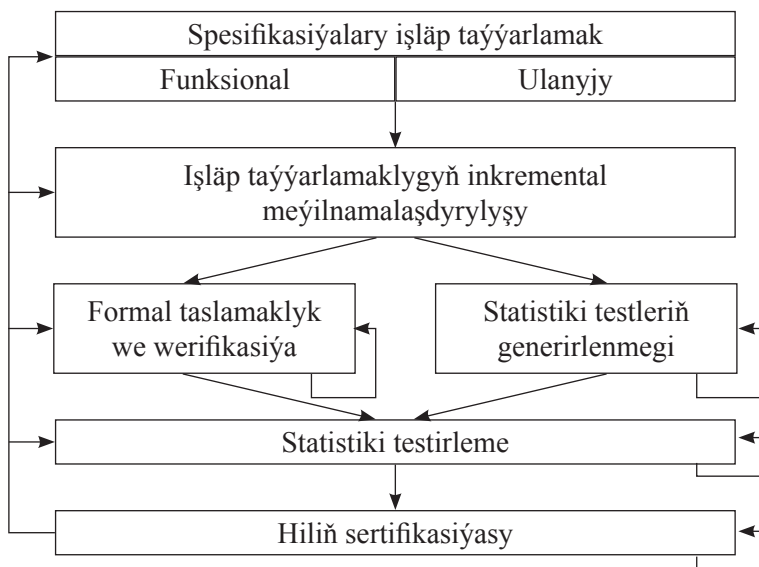
Çemeleşmeleriň bu toparý programma üpjünçiligini döretmek prosesine maksimal formal talaplary saklaýar.

28.1. Arassalanan sehiň tehnologiýasy

Arassalanan sehiň tehnologiýasynyň (cleanroom process model) esasy ideýalary Harlan Mills tarapyndan 80-nji ýyllaryň ortalaryna hödürlendi. Bu tehnologiýa indiki bölümlerden düzülýär (28.1-nji surat):

- funksional we ulanyjy spesifikasiýalaryň işlenilmegi;
- işläp taýýarlamaklygyň inkremental meýilnamalaşdyrylyşy;
- formal werifikasiýa;
- statistiki testirleme.

Taslamaklyk prosesi, programmanyň «Ýaşşik» diýip atlandyrylýanlardan düzülen funksiýa görnüşinde görkezilişi bilen baglanyşykly:



28.1-nji surat. Arassalanan sehiň tehnologiýasy

- fiksirlenen argumentli (höweslendirmeler) we netijeli (jogaplar) *gara ýaşşik*;
- içki ýagdaýy bellenilýän *ýagdaýly ýaşşik*;
- amala aşyrylyşy ädimleýin anyklaýyşda funksiýalaryň toplumy görnüşinde görkezýän *ak ýaşşik*.

Ýaşşikleriň ulanylyşyny indiki üç düzgünler kesgitleýärler:

- taslamak prosesinde kesgitlenen ähli maglumatlar ýaşşiklerde gizlenýärler (inkapsulirlenen);
- ähli prosesler ýaşşikleri parallel ýa-da yzygider ulanýanlar hökmünde kesgitlenen;
- her bir ýaşşik ulgam iýerarhiýasynda bellibir ýeri eýeleýär.

Bu tehnologiýada programmany alyp barmak ýaly proses ýok diýen ýaly. Ony formal werifikasiýa prosesi çalyşýar. Her bir dolandyryjy struktura üçin degişli korrektlik barada şert barlanýar.

Arassalanan sehiň tehnologiýasy toparlaýyn işlemekligi göz önünde tutýar, diýmek, testleriň taslanylyşy, anyklanylyşy, inspektsiýasy we taýýarlanylyşy dürli adamlar bilen amala aşyrylýar.

28.2. Formal genetiki çemeleşmeler

Bu günki gün subut edijilik häsiýete eýe bolan programmirlenme usullar dörediler. Şeýle usullaryň üçüsi eýýäm seredilen genetiki çemeleşmelere degişlidir. Emma olarda formal, matematiki spesifikasiýalar hasaba alnan:

- *formal sintezirleýän programmirlenme* – logiki formulalalaryň toplumy matematiki spesifikasiýasyny ulanýar. Bu programmirlenmegiň iki görnüşi bar:
 - *logiki*, onda programma teorema ýaly düşünilýän spesifikasiýadan konstruktiv subutnama hökmünde alynýar;
 - *transformasion*, onda spesifikasiýa programma görä deňleme hökmünde seredilýär we simwoliki özgertmeler bilen programma özgerýär;
- *formal ýygnama programmirlenme* spesifikasiýany eýýäm belli bolan kompozisiýa hökmünde ulanýar;
- *formal anyklaýjy programmirlenme* garyşyk hasaplaşyklar we annotasiýalar boýunça anyklamak ýaly çemeleşmeleri ulanýar.

Formal genetiki çemeleşmeleriň iň irki döwrebap işleriniň biri hökmünde B-tehnologiýasyny bellemek bolar. Onuň esasynda «ME-TEOR» Pariž metropoliteni dolandyrmak ulgamy işlenilip düzüldi.

29. Programmany çalt işläp taýýarlamaklygyň irki tehnologiiki çemeleşmeleri

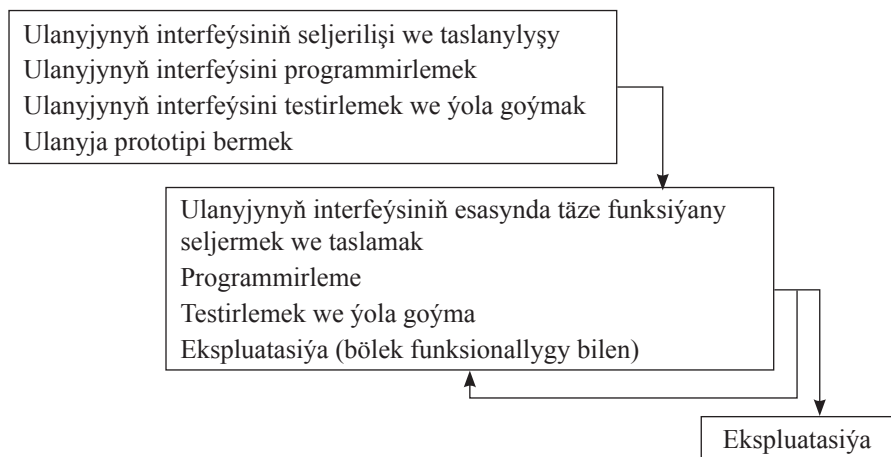
Kaskad çemeleşmelere alternatiwa bolup programmalary çalt işläp taýýarlamaklygyň irki tehnologiiki çemeleşmeleri çykyş edýärler. Bu çemeleşmeleriň ählisini indiki esasy aýratynlyklar birleşdirýärler:

- prototipiň iterasion işlenilmegi;
- buýrujy bilen ýakyn aragatnaşyk.

29.1. Ewolýusion prototipirleme

Ewolýusion prototipirleme (evolutionary prototyping) ilkinji prototip, adatça, ösen ulanyjy interfeýsini döretmekligi öz içine alýar. Ol şol badyna buýrujydan seslenme we mümkin bolan korrektiwalary almak üçin buýruja görkezilip bilner. Esasy başlangyç ünsi ulgamyň ulanyja ýüzlenen tarapyna berilýär.

Ewolýusion prototipirlemegi, haçanda buýrujy işläp taýýarlamaklygyň başlangyç tapgyrlarynda programma önümine edilýän



29.1-nji surat. Ewolýusion prototipirleme

talaplary doly we anyk düşündirip bilmedik ýagdaýynda, ulanmak bolýar.

Bu çemeleşmäniň düýpli ýetmezçiligi bolup taslamanyň dowamlylygy we bahasyny kesgiltäp bolmaýanlygy çykyş edýär. Programma önümini işläp taýýarlamaklygyň ewolýusion prototipirleme çemeleşmesi 29.1-nji suratda getirilýär.

29.2. Iterativ işläp taýýarlaýyş

Iterativ işläp düzmekligiň (iterative delivery) ilkinji prototipi eýýam ulgamyň özenini saklamaly. Şeýlelikde, onda eýýam funksionallygyň uly bölegi jemlenen. Indiki iterasiýalar ulanyja onuň interfeýsini, ulgam bilen generirlenýän habarlary we beýleki çykyş maglumatlary gutarnykly görnüşde kesgitlemäge kömek bermeli.

Ulgamyň özenine degişli bolmadyk ähmiýetli däl funksionallygy goşmaklyk rugsat edilýär.

Ýene-de bir gezek çalt işläp taýýarlamaklygyň seredilen iki sany usullarynyň arasyndaky tapawutlary belläp geçeliň. Iterativ işläp taýýarlaýyş ulgamyň özenini döretmekden başlanýar we soňra ony detallaşdyrýar. Ondan tapawutlylykda ewolýusion prototipirleme ulanyjynyň başlangyç interfeýsini döretmeklige we onuň esasynda funksionallygy goşmaklyga gönükdirilen.

Ulgamyň özeniniň seljerilişi we taslanylyşy
Ulgamyň özenini programmirlemek
Ulgamyň özenini testirmek we ýola goýmak
Ulanyja prototipi bermek

Ulanyjynyň interfeýsini we özeniň detalizasiýasynyň esasynda möhüm däl funksionallygyny seljermek we taslamak
Programmirleme
Testirmek we ýola goýma
Ekspluatasiýa (bölek funksionallygy bilen)

Ekspluatasiýa

29.2-nji surat. Iterativ işläp taýýarlaýyş

29.3. Döwürleýin işläp taýýarlaýyş

Döwürleýin işläp taýýarlaýyş (staged delivery) öňki iki çemeleşmeleriň kemçiliklerini – taslamany tamamlamaklygyň möhletini kesgitläp bilmezligi aradan aýyrmaklyk üçin niýetlenen. Programma önümini işläp taýýarlamaklygy başlamazdan, döredilýän programma önüminiň nähili bolmalydygyny biz gowy bilýäris. Döwürleýin işläp taýýarlamaklygyň esasy meselesi – buýruja işjeň ulgamy boldugyndan irräk taýýar edip bermek. Ondan soň buýrujy täze funksiýany goşup, ulgamyň täze görnüşini alyp biler. Emma her döwürň soňunda alnan görnüşleriň her biri işjeň bolup durýar.

Bu çemeleşme her bir döwürň ahyrynda alynýan ulgamyň jikme-jik we düýpli testirlenmegini talap edýär.



Adaptiw tehnologiكي çemeleşme

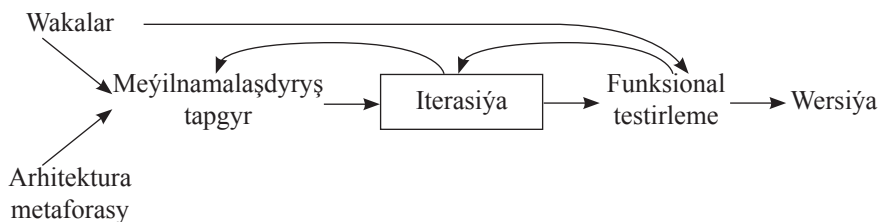
Adaptiw tehnologiكي çemeleşmeler üygemeleri goldaýan çemeleşmeler hökmünde döredildiler. Olar özünde üýtgemeler geçýän bolsa-da, üýtgemelerden diňe utýarlar. Bu çemeleşmeler proses üçin däl-de, adam üçin gönükdirilendir.

30.1. Ekstremal programmirleme

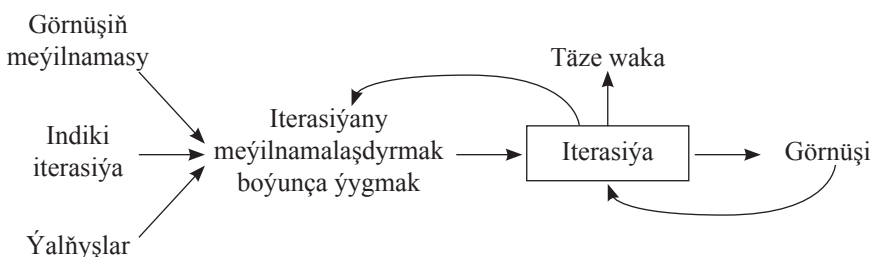
Programmalary çalt işläp taýýarlamaklygyň ideýalary *ekstremal programmirleme (extreme programming)* çemeleşmesinde has konsentrirlenen boldular.

Programmalary çalt işläp taýýarlamak çemeleşmesine mahsus bolan iki sany esasy häsiýet bu çemeleşmede hem esasy bolup durýarlar. Bu çemeleşmede jemlenen usullar özbaşdak täze däl, emma olaryň rasional birleşmesi we bilelikdäki ulanylyşy düýpli netijeleri we üstünlik bilen amala aşyrylan taslamalary berýär.

Çemeleşme döredilýän ulgamyň seljerilişini geçirmekden we ilkinjilik funksionallygy kesgitlemekden başlanýar. Netijede, wakalaryň, ulgamyň mümkin bolan ulanmalarynyň sanawy düzülýär. Her bir waka san görkezijileri bilen bahalandyryp bolýan bellibir meselelere gönükdirilen bolmaly. Umuman, wakanyň gymmatlylygy



30.1-nji surat. Ekstremal programmirlemek esasynda taslamanyň üstünde işlemek



30.2-nji surat. Iterasiýa

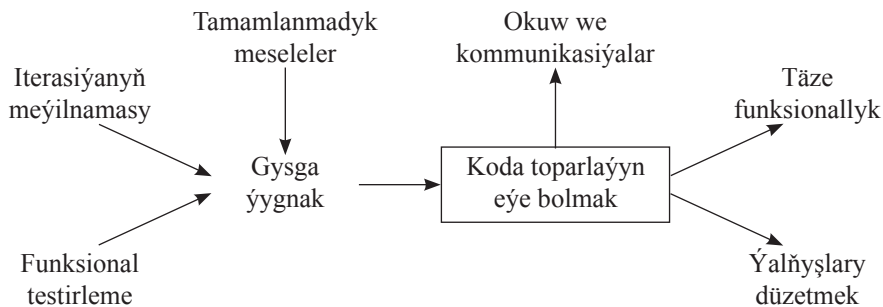
ony işläp taýýarlaýjy topar bilen döretmek üçin material we wagt çyk-dajylary bilen kesgitleýär.

Buýrujy taslama üçin bähbitlilikine we gymmatlylygyna esaslanyň, indiki iterasiýa üçin wakalary saýlaýar. Programmanyň ilkinji görnüşi üçin buýrujy logiki baglanyşykly has möhüm wakalaryň uly bolmadyk sanyny kesgitleýär. Her bir indiki görnüşi üçin galanlarynyň içinden has wajyp wakalar saýlanylýarlar (30.1-nji surat).

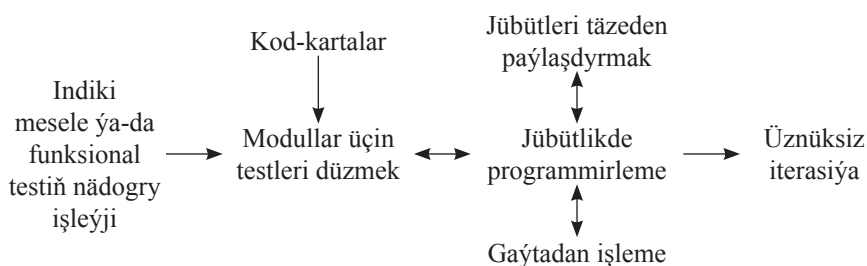
Bu çemeleşmäniň düýpli usullarynyň biri bolup funksional testirleme çykyş edýär. Testirleme prosesiniň iki sany aýratynlyklary bar:

- programma düzüjiler programmany testirmek üçin testleri özlari ýazýarlar;
- bu testler kodlamakdan öň ýazylýalar.

Her bir aýratyn testiň dogry işleýşine ynanmaklyk ýuwaş-ýuwaşdan umuman ulgamyň dogry işlemegine ynanýrýar.



30.3-nji surat. Işläp taýýarlaýyş



30.4-nji surat. Koda toparlaýyn eýelemek

Her bir iterasiýanyň maksady (30.2-nji surat) – programmanyň täze görnüşine täze wakalary goşmak. Meýilnamalaşdyrmak boýunça ýygnaqda wakalaryň haýsylary we nädip amala aşyrylmalydygy kesgitlenýär.

Işläp düzüji toparyň ählisi programma koduna eýe bolany üçin (30.3-nji surat), eger-de ol zerur hasap etse her bir programma düzüjä islendik wagt ulgamyň kodunyň islendik bölegini kämilleşdirmäge mümkinçilik berýär. Programma düzüji bellibir meseleleriň amala aşyrylyşyny öz üstüne alýar. Işlenilýän mesele boýunça soraglaryň ýüze çykan ýagdaýynda buýrujynyň gatnaşmagynda gysga (15 minut) ýygnaq geçirilip bilner.

Meseläni amala aşyrmak üçin oňa jogapkär programma düzüji özüne ýoldaş tapmaly (30.4-nji surat). Iň soňky kod bir iş stansiýasynda iki programma düzüjiler bilen ýazylýar.

30.2. Adaptiw işläp taýýarlaýyş

Adaptiw işläp taýýarlaýyş (Adaptive Software Development – ASD) çemeleşmäniň esasynda üç çyzykly däl biri-biriniň üstüni ýapýan fazalar: (pikirlenmek, hyzmatdaşlyk we okuw) goýulandyr.

Adaptiw işläp taýýarlaýyşda meýilnamalaşdyrmaklygyň netijeleri önünden belli däl. Adaty meýilnamalaşdyrmaklykda meýilnamadan gyşarma soňy bilen düzedilýän ýalňyşlyk bolup durýar. Bu çemeleşmede gyşarmalar dogry çözümlere getirýärler.

Programma düzüjiler adaptiw işläp taýýarlaýyş çemeleşmede kesgitsizligi ýeňip geçmek üçin özleriniň arasynda aktiw işlemeli. Taslamalaryň ýolbaşçylary programma düzüjileriň arasynda gowy kommunikasiýalary üpjün etmelidir.

Okatmaklyk – çemeleşmäniň hemişelik we wajyp häsiýetnamasydyr. Programma düzüjiler hem, buýrujylar hem iş prosesiniň dowamynda özüniň wezipe-borçlaryna we meýilnamalaryna täzeden seretmeli. İşläp taýýarlamaklygyň her bir sikliniň netijeleri indikiniň taýýarlanyşynda ulanylýarlar.

30.3. Crystal tehnologiki çemeleşmeleriň maşgalasy

Crystal tehnologiki çemeleşmeleriň maşgalasynyň döredijisi – Alistr Koubern. Çemeleşmeleriň her biri iki sany: «Güýçlendirilen kommunikasiýa» we «Ýeňilleşdirilen iş önümler» bazalaýyn düzgünlerinde düzülýär. Bu düzgünleriň ikisini-de islendik taslamada aňsatlyk bilen ulanmak we görnüşini üýtgetmek bolýar.

Crystal çemeleşmede taslamanyň ýerini üç ölçegli giňişlikde indiki ölçegler boýunça kesgitlemek kabul edilen:

- *taslamanyň tankydylygy*. Bu ulgamda emele gelen defektiň howplulyk derejesi. Adatça, howplulygyň dört: adamyň janyna howp salmak, uly möçberli pullary ýitirmek, uly däl möçberli pullary ýitirmek, işde rahatsyzlyk yzygiderliligini belleýärler;
- *taslamada iş bilen üpjün edilen adamlaryň sany*. Yzygiderlilik üçin bu ýerde 2-den 2,5-e çenli ösüş koeffisiýenti ulanylýar.

Işläp düzüjileriň sanyny iki esse köpeltmekligiň netijesinde goşmaça ylalaşylyk we goşmaça aragatnaşyk mehanizmleri işläp düzmekligi talap edýär;

- *prosesi guramaklygyň netijesinde ýetmeli maksatlar*. Bu maksatlar netijelilik kriteriler bolup çykyş edýärler. Şeýle maksatlar bolap indikiler:
 - önümiň ýygy-ýygýdan üýtgäp durýan talaplaryna laýyk gelmekligi;
 - öndürjilikligi we talabalaýyk hilinde önümi taýýarlamak wagtyny ýokarlandyrmak;
 - önümiň ýokary hiline güwa geçmek bolup biler.



31. Barlag programmirlmeginiň çemeleşmeleri

Barlag programmirlmeginiň indiki aýratynlyklary bar:

- işläp düzüji gözleg ugruny anyk bilýär, emma önde goýlan maksada nähili ýetjekdigini öňünden bilmeýär;
- şol ýa-da beýleki netijäni almak üçin resurslaryň göwrümini görmek mümkinçiligi ýok;
- işläp taýýarlaýşy detallaýyn meýilnamalaşdyrmak bolmaýar;
- iş anyk ýerine ýetirijiler bilen bagly we olaryň şahsy hasiýetlerini görkezýär.

Beýleki çemeleşmelere garanyňda barlag programmirlmeginiň esasynda sungat ýatyr. Bu sanalyp geçilen aýratynlyklaryň çylşyrymly meseleleri çözmekligiň hakyky ýagdaýyny beýan edýändigleri bilen düşündirilýär.

31.1. Kompýuter darwinizmi

Bu çemeleşmäniň atlandyrylyşy Ken Tompson tarapyndan berildi. Çemeleşme işläp taýýarlamaklygyň beýgelýän düzgünine esaslan. Ulgam taslamanyň irki stadiýalarynda döredilýän we soň hemişe modifisirlenip durýan esasy komponentleriň we programmalaryň daşynda gurulýar. Has uly bloklar öň döredilen kiçi bloklardan düzülýärler.

Kompyuter darwinizmi intensiw testirlemekde esaslanan synaglar we ýalňyşlar usulyny emele getirýär. Ol işläp düzüjileriň düzen minimal görnüşi bolsa-da, özi hem islendik döwründe programma işläp durmaly. Tebigy saýlap-seçme diňe ýaşamaga ukyplylary goýýar.

Çemeleşme üç sany esasy proseslerden:

- Maketirlemeden (prototipirleme);
- Testirlemeden;
- Ýola goýmadan ybarat.

31.2. Fragmentar programmirleme

Bu çemeleşmäniň aýratynlygy – ilki başda işleýän böleklerden (fragmentlerden) düzülen programmanyň şablony döredilýär. Soňra kem-kemden ahyrky maksada ýetmeklik amala aşyrylýar. Çemeleşme, haçanda meseläniň köp bölegine talaplar anyk kesgitlenmedik bolsa ulanylyp bilner. Birleşdiriji şablon eýýäm ýazylan fragmentleriň baglanyşygyny gurnaýan «draýwer»-programmanyň ornuny ýerine ýetirýär.

PEÝDALANYLAN EDEBIÝATLAR

1. *Gurbanguly Berdimuhamedow*. Türkmenistanda saglygy goraýşy ösdürmegiň ylmy esaslary. Aşgabat. Türkmen döwlet neşirýat gullugy, 2007.
2. Türkmenistanyň Ministrler Kabinetiniň ýanyndaky Baş arhiw müdirligi. Türkmenistanyň Prezidentiniň Arhiw gaznasy. «Türkmenistanyň Prezidenti Gurbanguly Berdimuhamedow». Gysgaça terjimehal. Aşgabat, 2007.
3. *Gurbanguly Berdimuhamedow*. Garaşsyzlyga guwanmak, Watany, halky söýmek bagtdyr. Aşgabat. Ylym, 2007.
4. *Gurbanguly Berdimuhamedow*. Ýurdy täzedan galkyndyrmak baradaky syýasaty. Aşgabat. Türkmen döwlet neşirýat gullugy, 2007.
5. *Gurbanguly Berdimuhamedow*. Türkmenistan – sagdynlygyň we ruhubelentligiň ýurdy. Aşgabat. Türkmen döwlet neşirýat gullugy, 2007.
6. *Gurbanguly Berdimuhamedow*. Eserler ýygyny. Aşgabat. Türkmen döwlet neşirýat gullugy, 2007.
7. *Gurbanguly Berdimuhamedow*. Täze galkynyş eýýamy. Wakalaryň senenamasy – 2007-nji ýyl. Aşgabat. Türkmen döwlet neşirýat gullugy, 2008.
8. Türkmenistanyň Prezidenti Gurbanguly Berdimuhamedowyň daşary syýasaty. Wakalaryň hronikasy. Aşgabat, 2007.
9. Türkmenistanyň Prezidentiniň obalaryň, şäherçeleriň, etrapdaky şäherleriň we etrap merkezleriniň ilatynyň durmuş-ýaşayyş şertlerini özgertmek boýunça 2020-nji ýyla çenli döwür üçin Milli maksatnamasy. Aşgabat, 2007.
10. *Gurbanguly Berdimuhamedow*. Osüşň täze belentliklerine tarap. Saýlanan eserler. I-VIII tom. Aşgabat. Türkmen döwlet neşirýat gullugy, 2008–2015.

11. *Gurbanguly Berdimuhamedow*. Döwlet adam üçindir. Aşgabat. Türkmen döwlet neşirýat gullugy, 2009.
12. *Gurbanguly Berdimuhamedow*. Türkmenistanyň ykdysady strategiýasy: halka daýanyp, halkyň hatyrasyna. Aşgabat. Türkmen döwlet neşirýat gullugy, 2010.
13. *Gurbanguly Berdimuhamedow*. Türkmenistanyň durmuş-ykdysady ösüşiniň döwlet kadalaşdyrylyşy. I tom. Ýokary okuw mekdepleriniň talyplary üçin okuw gollanmasy. Aşgabat. Türkmen döwlet neşirýat gullugy, 2010.
14. *Gurbanguly Berdimuhamedow*. Türkmenistanyň durmuş-ykdysady ösüşiniň döwlet kadalaşdyrylyşy. II tom (Goşundylar). Ýokary okuw mekdepleriniň talyplary üçin okuw gollanmasy. Aşgabat. Türkmen döwlet neşirýat gullugy, 2010.
15. Türkmenistanyň durmuş-ykdysady ösüşiniň 2011–2030-njy ýyllar üçin Milli maksatnamasy. Aşgabat. Türkmen döwlet neşirýat gullugy, 2010.
16. *Gurbanguly Berdimuhamedow*. Älem içre at gezer. Roman. Aşgabat. Türkmen döwlet neşirýat gullugy, 2011.
17. Türkmenistany 2012–2016-njy ýyllarda durmuş-ykdysady taýdan ösdürmegiň maksatnamasy. Aşgabat. Türkmen döwlet neşirýat gullugy, 2012.
18. Türkmenistanyň Prezidentiniň namalarynyň we Türkmenistanyň Hökümetiniň çözgütleriniň ýygındysy. 1991–2015 ý.ý.
19. *T. Jurdekow*. Ata arzuwyny amala aşyrýan Agtyk. Aşgabat, 2009.
20. *B.M. Iwanow, H.D. Ýazhanowa, A.Ý. Ýazgylýjow, G.G. Seyithanowa, O.G. Atdaýewa, Ş.A. Kiçiyewa, G.A. Şükürow, G.I. Işkabulow, D.D. Ilmyradow*. Ykdysadyýetde awtomatlaşdyrylan informasion tehnologiýalar. Aşgabat. Türkmen döwlet neşirýat gullugy, 2001.
21. *A. Ýazgylýjow, A. Ýazgulyýew, H. Ýazhanowa, G. Şükürow, Ýu. Aronskiý, B. Iwanow, G. Seyithanowa, O. Atdaýewa, Ş. Kiçiyewa, H. Ýagmyrow, N. Başimowa*. Informatika we kompýuter tehnikasynyň esaslary. Aşgabat. Türkmen döwlet neşirýat gullugy, 2001.
22. *O.G. Atdaýewa, G.G. Seyithanowa, G.A. Şükürow, D.D. Ilmyradow, A.K. Myradow*. Hasaplaýyş maşynlar, ulgamlar we torlar. Aşgabat. Türkmen döwlet neşirýat gullugy, 2001.

23. *G. Işkabulow, O. Atdaýewa, G. Gylyçnyýazowa, H. Ýagmyrow.* Programmirlemegiň häzirki zaman tilsimatlary. Aşgabat. Türkmen döwlet neşirýat gullugy, 2001.
24. *G. Işkabulow, B.M. Iwanow, O.G. Atdaýewa, A.K. Myradow, M. Gylyçnyýazowa, G.G. Seyithanowa.* Maglumat bazalaryny taslamak. Aşgabat. Türkmen döwlet neşirýat gullugy, 2001.
25. *M. Babakulyýew, Ö. Muhammetberdiýew.* Maglumatlar tehnologiýalarynyň adalgalarynyň sözlügi. Aşgabat. Ylym, 2004.
26. *D. Kulyýew, T. Arazmyradow, M. Berenow, N. Garyagdyýew.* Kompýuterde işlemek. Aşgabat. Türkmen döwlet neşirýat gullugy, 2005.
27. *A. Ýazgylýjow, B. Iwanow.* Kompýuter tehnologiýalary, Aşgabat. Ylym, 2008.
28. *Ç. Aşyralyýew.* Kompýuter tehnologiýalary. Aşgabat. Türkmen döwlet neşirýat gullugy, 2008.
29. *S. Roger. Pressman.* Software engineering. High Education. 2010
30. Гибкая методология разработки программного обеспечения. Microsoft® Solutions Framework. 2010.
31. *И.О. Одинцов.* Профессиональное программирование. Системный подход. – 2-е издание. СПб.: БХВ-Петербург, 2004 г.
32. *С.А. Орлов.* Технологии разработки программного обеспечения. Разработка сложных программных систем. СПб.: Питер, 2011.
33. *Иан. Соммервилл.* Инженерия программного обеспечения. «Вильямс», 2010.
34. *Э. Гамма, Р. Хелм, Р. Джонсон, Д. Влиссидес.* Приемы объектно-ориентированного проектирования. СПб.: Питер, 2010.
35. *Internet saýtlary:*
<http://www.turkmenistan.gov.tm>
<http://www.vs2010.ru>
<http://univer-nn.ru/it/etapiAIS.php>
<http://www.interface.ru>
<http://www.microsoft.com>
<http://www.intuit.ru>
<http://www.w3schools.com>

MAZMUNY

Giriş.....	7
------------	---

I. Programmirlemegiň esasy düşünjeleri

1. Programmirleme we onuň bölümleri	9
1.1. Programmirleme – informatikanyň bölegi we onuň bölümleri hökmünde	9
1.2. Programmirlemegiň inženeriýasy	10
2. Programmirlemegiň esasy düşünjeleri we kesgitlemeleri	12
2.1. Programma we programma önümi	12
2.2. Algoritmeler	13
2.3. Modeller we modelirleme	14
2.4. Programmanyň ýaşayyş sikli	15
2.5. Platformalar	16
3. Taslamalary işläp taýýarlamak prosesi	17
3.1. Taslamalar we olaryň häsiýetleri	17
3.2. Taslamalary işläp taýýarlamak prosesi we oňa gatnaşyjylar	19
4. Professional programmirlemegiň esasy meselesi	22

II. Programmirlemegiň usulyýeti

5. Programmirlemegiň usulyýetiniň esasy düşünjeleri we olaryň klassifisirilenmegi	24
5.1. Programmirlemegiň usulyýetiniň esasy düşünjeleri we kesgitlemeleri	24
5.2. Programmirleme usulyýetleriň klassifikasiýasy	26
6. Imperatiw programmirlemegiň usulyýeti	28
7. Obýekte gönükdirilen programmirleme usulyýeti.....	31
8. Funksional programmirlemegiň usulyýeti.....	35
9. Logiki programmirlemegiň usulyýeti.....	38

10. Çäklendirmelerde programmirlemegiň usulyýeti	41
11. Düzümleýin imperatiw programmirlemegiň usulyýeti.....	43
12. Imperatiw parallel programmirlemegiň usulyýeti	45
13. Logiki parallel programmirlemegiň usulyýeti	48

III. Programma üpjünçiligini döretmekligiň tehnologiki prosesleri

14. Programma üpjünçiligini döretmekligiň tehnologiýalarynyň esasy düşünjeleri.....	50
15. Programmany işläp düzmekligiň umumy düzgünleri.....	51
16. Programma üpjünçiligini döretmekligiň tehnologiki prosesleriň klassifikasiýasy	53
17. Programma önümini döretmek ideýasynyň döreýşi we onuň derňewi	55
18. Programma önüminiň taslamasyny dolandyryş prosesi	58
18.1. Taslamanyň dolandyrylyşynyň usullary	58
18.2. Taslamanyň meýilnamalaşdyrylyşy	59
18.3. Işleri paýlaşdyrmak.....	60
19. Talaplaryň seljerilişi we programma önüminiň taslanylyşy	60
19.1. Talaplaryň seljerilişi, programma önüminiň taslanylyşy we arhitekturasy	60
19.2. Arhitekturanyň taslanylyşy («uludan» taslamak).....	62
19.3. Modullaryň taslanylyşy («kiçiden» taslamak).....	63
20. Programma önüminiň programmirlenilmegi (amala aşyrylyşy).....	65
21. Programma önümlerini testirmek we ýola goýmak.....	72
21.1. Programma önümleriniň testirlenilişi	72
21.2. Programma önümlerini ýola goýmak	75
22. Programmany işe girizmek	76
23. Programma önüminiň ekspluatasiýasy we alnyp barylýşy.....	78

IV. Programma üpjünçiligini döretmekligiň tehnologiki çemeleşmeleri

24. Programmirleme tehnologiýalarynyň çemeleşmeleriniň klassifikasiýasy	82
--	----

25. Kaskad görünüşli tehnologiki çemeleşmeler	83
26. Karkas görünüşli tehnologiki çemeleşmeler.....	88
26.1. Rasional unifikirlenen proses	88
26.2. Microsoft Solution Framework prosessorlarynyň modeli	89
27. Genetiki tehnologiki çemeleşmeler	91
27.1. Sintezirleýän programmirlеме	91
27.2. Ýygnama (giňeldilýän) programmirlеме	92
27.3. Anyklaýjy programmirlеме.....	94
28. Formal özgertmeler esasyndaky çemeleşmeler	95
28.1. Arassalanan sehiň tehnologiýasy	95
28.2. Formal genetiki çemeleşmeler	96
29. Programmany çalt işläp taýýarlamaklygyň irki tehnologiki çemeleşmeleri	97
29.1. Ewolýusion prototipirlеме	97
29.2. Iterativ işläp taýýarlaýyş.....	98
29.3. Döwürleýin işläp taýýarlaýyş	99
30. Adaptiw tehnologiki çemeleşme.....	99
30.1. Ekstremal programmirlеме	99
30.2. Adaptiw işläp taýýarlaýyş.....	102
30.3. Crystal tehnologiki çemeleşmeleriň maşgalasy	102
31. Barlag programmirlемeginiň çemeleşmeleri	103
31.1. Kompýuter darwinizmi	103
31.2. Fragmentar programmirlеме	104
Peýdalanylan edebiýatlar	105

***Gülmar Bäşimowa, Bäşim Iwanow, Oguljan Atdayewa,
Berdimyrat Orazow***

**AWTOMATLAŞDYRYLAN
ULGAMLARYŇ WE KOMPÝUTER TEHNIKASYNYŇ
PROGRAMMA ÜPJÜNÇILIGI**

I kitap

Ýokary okuw mekdepleri üçin okuw kitaby

Redaktor

O. Aşyrowa

Surat redaktory

G. Orazmyradow

Teh. redaktor

O. Nurýagdyýewa

Neşir üçin jogapkär

Çap etmäge rugsat edildi. Ölçeği 60x90¹/₁₆.
Edebi garniturasy. Şertli çap listi. Şertli reňkli ottiski.
Hasap-neşir listi. Çap listi 7,0. Sargyt №400. Sany.

Türkmen döwlet neşirýat gullugy.
744000. Aşgabat. Garaşsyzlyk şaýoly, 100.

Türkmen döwlet neşirýat gullugynyň Metbugat merkezi.
744015. Aşgabat. 2127-nji (G. Gulyýew) köçe, 51/1.