

## Дәріс 2 Объектілі-бағытталған талдау мен жобалаудың қазіргі технологиясы

### 2.1 Объектілі-бағытталған талдау мен жобалаудың концепциясы

Түсініктемелерді анықтау: программа, деңгейлері және категориялары (бағыттары) программалау, программаны өңдеу және аспаптары. Аспаптар, ол жұмысты орындау үшін арналған құралдар, яғни программаны өңдеу және тарату екі топқа бөлінеді аппараттық және программалық. Аппараттық – микропроцессорлар және қосылатын (сыртқы) құрылғылар. Программалық – олар жобалау методологиясымен анықталған, барлық жұмыстарды орындауға мүмкіндік беретін программалар. Беріліп отырған пән негізінен, компьютерге программаларды орнату және өңдеу үшін қолданылатын программалық аспаптық құралдарды оқып үйренуге арналған. Программалық өнімді (ПӨ) өңдеу көптеген бір-бірімен байланысқан әрекеттерден тұрады, олар:

- деректер моделін құру және есептеу әдістемесі;
- есептеуді қамтамасыздандыратын, функциональдық сипаттамасы;
- деректер құрылымын анықтау – компьютерде және алгоритмде көрсетілу моделі;
- есепті тарату әдістерін сипаттау және анықтау (тестер және шешу алгоритмі);
- пайдаланушы интерфейсін сипаттау және анықтау;
- ПӨ қолдау құралдарын анықтау;
- есеп спецификациясы;
- тестілеу программасын қоса отырып, программа текстін жазу;
- программаның тестіленуі, трансляциясы және жөнделуі;
- қолдау кітапханаларын қосу және байланыстыру;
- орындау ортасын құру; орындамалық модульді орналастыру және жүктеу;
- орнатылған көмекті құру және өңдеуді құжаттау;
- орнатылатын (инсталляциялық) ПӨ пакетін құру.

Rational Unified Process (RUP) аймағында программаларды өңдеуге арналған әрекеттер жиыны келесі этаптардан тұрады: - талаптарды анықтау; - жобалау; - программалау; - тестілеу; -ендіру.

Көрсетілген жұмыстарды орындау үшін көптеген программалар жиынтығы үнемі өңделіп және толықтырылып отырады – ол аспаптар, программаның өңделу процессін автоматтандыру және қалыптастыру үшін мүмкіндік береді. Бұл құралдарды қолдану өңдеу және программалық өнімді енгізу уақытын азайтады.

Компьютер үшін программа – бұл программалау тілінде құрылған немесе жазылған, тәртіптер тізімі. Программалау тілі берілген есепті шешу үшін және тиімді, жылдам, сапалы, экономды жобалау мақсатында таңдалады.

Тіл деңгейі – программалау деңгейі. Тіл деңгейі – программалау деңгейі. Төменгі деңгейлі программалау – бұл микропроцессор деректер форматын және командаларын кодтық немесе мнемоникалық формада (ассемблер) қолданатын программалау. Проблемалы – бағытталған программалау (текстуальды) – бұл

деректер форматын және командаларын проблемалы – бағытталған тілдерде программалау. Визуальды программалау – бұл деректер форматын және командаларын программаларды өңдеудің визуальды құралдары арқылы программалау. Объектілік немесе компоненттік программалау әртүрлі конструктивтерді бірлік элемент ретінде қолдануға мүмкіндік береді.

Командалық (атомарлы), құрылымдық, модульдік программалау – категориялары, программаларды өңдеу кезінде қолданылатын тілдің типтік конструкциясымен анықталады. Құрылатын программа үшін, проблемалы аймақ, программалаудың бағытын анықтайды – ғылыми түрде, бизнес есептерін шешу үшін, объектілер мен процесстерді басқару үшін, ақпараттарды көрсету және басқару үшін, Интернет пен қарым – қатынас жасау үшін, деректер қорымен қарым – қатынас жасау үшін және т.б. қолданылады.

Құрылымдық программалау концепциясы.

«Құрылымдық программалау – программалау методологиясы, жүйелік түрдегі талдауға негізделген, ПҚ тарату және жобалау. 70 – жылдардың басында пайда болды және соншалықты өміршең, сондықтан да осы күнге дейін пайдаланылады. Бұл технологияның негізі келесі тәртіптерден тұрады:

- Қиын есептер кішірек бөліктерге бөлінеді, функциональды түрде негізінен басқарылатын есептер дұрыс. Әрбір есептің бір кірісі және бір шығысы болады. Бұл жағдайда программа ағымы көптеген есептер тобынан және түсінікті функционалды тағайындамасынан құралады.

- Есепте қолданылатын, басқарылатын құрылымның қарапайымдылығы және атомарлығы. Бұл, логикалық есеп минималды, функционалды толық қалыптасқан қарапайым басқарылатын құрылымнан тұруы керек екендігін көрсетеді.

- Программаның өңделуі этап бойынша орындалады. Әрбір этапта шектелген есеп саны, нақты қойылған тапсырмасымен және есеп барысындағы оның атқаратын мәнімен, түсінікті қойылымы шешілуі керек. Егер ондай шамаға жетпесе, онда ол бұл этап өте үлкен және оны тағы басқа қадамдарға бөлу керек екендігін білдіреді.

Модульдік программалау концепциясы.

Құрылымдық технология сияқты мұнда да, модульдік программалау концепциясын бірнеше түсініктемелер және тәртіптер түрінде қалыптастыруға болады:

- Есептің функциональдық декомпозициясы – үлкен есепті кіші есептерге бөлу, олар функциональды өзбектінше орындалатын ішкіесептер - модульдер. Модульдер өзара кіріс және шығыс деректерімен байланысқан.

- Модуль - модульдік программалау концепциясының негізі. Әрбір модульдің функциональдық декомпозициясы бір кірісі және бір шығысы бар "қара жәшік" ретінде қаралады. Модульдік көзқарас программаның модернизациясын процесстің эксплуатациясы кезінде ешбір қатерсіз орындауға мүмкіндік береді және оны қолдап отырады. Қосымша модульдік программалау бір жобаның программасының әртүрлі бөліктерін әртүрлі программалау

тілдерінде орындауға мүмкіндік береді, одан кейін компоновкалау құралдары арқылы оларды бір жүктемелеу модуліне біріктіреді.

- Таратылатын шешімдер қарапайым және түсінікті болуы керек. Егер модульдің атқаратын қызметі түсініксіз болса, онда бастапқы немесе аралық декомпозиция есептері дұрыс, керек еткен деңгейде, сапалы жүрмегендігін білдіреді. Бұл жағдайда есепті қайта талдау керек, мүмкін, қайта бөлулер жасау, яғни ішкі есептерге бөлу жұмыстарын орындау керек болады. Жобада қиын орындалатын орындар кездесе, онда оларды жүйемен ойластырылған комментарилер арқылы құжаттау керек. Бұл процесс модульдердің жұмысының толық түсінікті, қызметтері анықталған және байланыстырылғанға дейін қайталана береді.

- Модульдің барлық айнаымалыларының қызметі комментерилердің көмегімен анықталуы және жазылуы керек.

## **2.2 Объектілі-бағытталған программалаудың методологиясы**

Объектілі-бағытталған парадигма.

ОБП идеясы негізінен осы деректерді өңдейтін процедураларды бір бүтін - объектіге байланыстыру үшін жасалған. ОБП маңызды үш принцип негізінде құрылған, олар объектілерге жаңа қасиеттер береді. Бұл принциптар ретінде инкапсуляция, мұрагерлену және полиморфизм алынады.

- Инкапсуляция** – Осы деректерді өңдеу алгоритмдері мен деректерді бір бүтінге біріктіру. ОБП деректері көлемінде – объект полесі, алгоритмдері – объектілік әдістер алынады. Полелер мен алгоритмдер іштен қолданылады, - жалпы, сырттан немесе тек объект ішінде – соған тәуелді, ішкілер немесе байланысқан объектілер – қорғалған.

- Мұрагерлену** – объектілер қасиеті өздерінің мұрагерлерін туындайды және оларға өздерінің қасиеттерін үнсіздікпен ұсынады. Объект - мұрагерленуші, ол қасиеттерді толықтыра алады немесе басқамен оны ауыстыра (орынбастыра) алады.

- **Полиморфизм** – туыстық объектілердің қасиеттері (яғни бір объектіден тараған бір түбірі бар объектілер) негізінен бір – біріне ұқсас, мәні жағынан бірдей проблемаларды әртүрлі әдістермен, оның уақытына және орналасу ретіне байланысты шешу.

Программаны өңдеу үшін алынған аспап таңдалған деңгей негізімен, бағытымен, өңдеу категориясымен анықталады және текстуальды немесе визуальды түрде беріледі. Қазіргі заманғы программалау – компоненттік (объектілік), уақиғалық және визуальды болып алынады. Түсініктемені анықтайтын болсақ.

Программаны өңдеу мемлекеттік және шетелдік стандартқа сәйкес, өңдеуші ұсынған технологиялар мен әдістемелер немесе типтік программалар арқылы орындалады.

Модельдің көмегімен біз керегі жоқ детальдарды қарастырмай-ақ бар назарымызды негізгіге аудара отырып, проблеманы оңайлатуымыз мүмкін. Абстракциялау мүмкіндігі – күрделік феноменін шешуге көмектесетін, адам интеллектісінің іргелі қасиеті. Мыңдаған жылдар барысында суретшілер, қол

өнершілер және құрылысшылар нақты шығармашылық ойларды жүзеге асыру алдында түрлі модельдерді жобалау талпыныстарын жасаған. Бұл құбылыстар программалық қамтаманы жасау индустриясын айналып өткен жоқ. Күрделі программалық жүйені құрастыру үшін, автор оның қасиеттерін түрлі көзқарастар тарапынан абстракциялауға, белгілеулердің нақты жүйелерінің көмегімен модельдерді жобалауға, олардың бастапқы талаптарға сәйкестігін тексеріп алуға міндетті, тек содан кейін ғана жүйені жаңа функциялармен толықтыра отырып, модельді практикада жүзеге асыруға болады.

**Негізгі әдебиеттер:** [1] – 59 б

**Қосымша әдебиеттер:** [2] –25б.

**Бақылау сұрақтары:**

1. ОБП деген не?
2. Модель дегеніміз не?
3. Бизнес аналитика деген не?
4. ОБП құру үшін қандай құрал түрлері қажет?
5. ОБП принцип түрлері?