

Мультимедиа компьютерлік технологияларын қолану арқылы оқу материалын баяндау аудиториялық сабақтар әр білім алушыға түсінікті болып қана қоймай, сонымен қатар уақытты үнемдеу мүмкіндігін береді. Компьютер және ақпараттық технологиялар арқылы жасалып жатқан оқыту процесі білімгердің жаңаша ойлау қабілетін қалыптастырып, оларды жүйелік байланыстар мен заңдылықтарды табуды игертіп, нәтижесінде – өздерінің кәсіби потенциалдарын қалыптастыруына жол ашуы сөзсіз.

Қолданылған әдебиеттер тізімі:

1. <http://www.ispring.ru/ispring-suite>.
2. ДузбаеваР.М. Формирование готовности студентов к интерактивному обучению. Алматы: Улагат, 2002.

Оспанова Ш.Б.¹, АманжоловаА.Р.²

1. *Ғылыми жетекші, жаратылыстану ғылымдарының магистрі, аға оқытушы*
2. *«Информатика» мамандығының 4 курс студенті, информатика және компьютерлік технологиялар кафедрасы*

BORLAND C++ BUILDER ПРОГРАММАЛАУ ТІЛІ – ЖОҒАРЫ ДЕҢГЕЙЛІК КҮРДЕЛІ ТІЛ

Адамзат дамуының жаңа кезеңі ХХІ ғасыр – бұл ақпараттық қоғам дәуірі, оның негізін компьютерлік технология құрайды. Сол себепті де бүкіл дүниежүзінде ең алдымен ақпараттық жаһандану үрдісі жаппай етек алууда. Оның таңғажайып көрінісі болса, «адам - компьютер - Интернет» байланыс жүйесі деп атап айтар едік.

ҚР-сында осы ақпараттық өркениет үрдісінен шығу үшін «Қазақстан 2030» стратегиясынан бастап, Елбасының дәстүрлі Жолдаулары бойынша барлық жағдайлар жасалып келеді. Әсіресе, білім беру саласы барынша ақпараттандыру (информатизация) мақсаты жүзеге асырылып отыр. Бұған қажетті заңнамалар мен бағдарламалар қабылданған.

Әлемдік өркениет жолмен бағытталған Елбасының «Жаңа әлемдегі жаңа Қазақстан» атты жолдауында: «Біз бүкіл еліміз бойынша әлемдік стандарттар деңгейінде сапалы білім беру қызметін көрсетуге қол жеткізуіміз керек» десе, ол үшін жаңа ақпараттық технологияларды оқытуды міндет етеді. Бір сөзбен айтқанда, басты мақсат білім сапасын арттыру арқылы әлемдік білім кеңістігіне кіре білу болады. Міне, сондықтан да қазірден бастап, «Объектіге бағытталған тілдер және программалау технология» пәндерінде әлемдік деңгейге жеткізе оқыту талабы қойылып отыр. Осыған сай айта кету қажет, егер «нені оқыту керек?» деген сауалға мемлекеттік білім стандарттары жауап беретін болса, «қалай оқыту керек?» деген мәселе мұғалімдер біліктілігімен тікелей байланысты.

Программалаудың парадигмалары: құрылымдық, модульдік және объектілі-бағытталған программалау

Бағдарламалауда парадигма деп бағдарламаның жұмысының ұйымдастырылуы түсініледі. Бағдарлама парадигмалары арасында мыналарын ерекше атап өтуге болады: құрылымдық, объектілі-бағытталған және модульдік. Көптеген тілдер бағдарламалаудың бірнеше парадигмаларын қолдайды. Сонымен қатар тек бір ғана парадигманы жүзеге асыруға бағытталған бағдарламалау тілдері де бар [1, 112 бет].

Құрылымдық бағдарламалау

Оларға жататын бағдарламалау тілдері: Fortran, Pascal, C. Директивті бағдарлама іс-әрекеттерді қадам бойынша бейнелеп нәтижеге қол жеткізуді көрсетеді. Сондықтан мұндай бағдарламалау түсінуге жеңіл болады. Құрылымдық бағдарламалауда командалардың орындалу реті кіріс деректеріне толығымен тәуелді болады. Директивалық бағдарламалауда кезінде кодтың бөліктерін ішкі бағдарламаларға(функция, әдістер) локализациялау концепциясы пайда болды. Және соларды кейін негізгі бағдарламаның әр түрлі жерінен шақырады. Шақыру кезінде ішкі бағдарламаға қайсібір деректер аргументтер түрінде берілуі мүмкін; ал ішкі бағдарлама өз кезегінде негізгі бағдарламаға нәтижені қайтара алады,яғни оның орындалуы барысында алынған деректер. Модульдік бағдарламалау модуль деп аталатын түсінік негізінде құрылған. Бұл – жеке бағдарламалық модульдер түрінде жасалған логикалық өзара байланысқан функционалды элементтер жиыны. Модуль сипаттайды:

1. Бір кіріс және бір шығыс – кірісте бағдарламалық модуль анықталған шығыс деректерінің жиынын алады, мазмұндық өңдеуді жүзеге асырады және нәтижелік деректерді жиынын қайтарады, яғни стандартты IPO(Input-Process-Output) –кіріс-процесс-шығыс принципі жүзеге асырылады.

2. Функционалды аяқталғандық–модуль басталған өңдеуді аяқтау үшін әрбір жеке функцияны толық құрамда жүзеге асыруда регламенттелген операциялар қатарын орындайды.

3. Логикалық тәуелсіздік–бағдарламалық модуль жұмысының нәтижесі шығыс деректеріне тәуелсіз, бірақ басқа модульдер жұмысының жұмысынан тәуелсіз.

4. басқа бағдарламалық модульдермен әлсіз ақпараттық байланыс–модульдер арасындағы ақпарат алмасу мүмкіндігінше минимизацияланған болуы керек.

5. өлшемі мен қиындығы бойынша қарастырылатын бағдарламалық элемент.

Модульдік бағдарламалау иерархиялық құрылымдарды қолдану мен компонент жүйенің қиындығы мен тәуелсіздікті қамтамасыз етумен күресудің екі ортақ тәсілдерін бағдарламаны құру процесі барысында жүзеге асырады. Бірінші тәсілді жүзеге асыру үшін ағаш түрдегі бағдарламаның модульдік құрылымын қолданады. Екінші тәсілді орындау үшін бағдарламалық модуль

қанағаттанатын белгілі бір талалаптар құрылады, яғни «жақсы» бағдарламалық модульдің негізгі сипаттамалары анықталады.

Объектілі-бағытталған тілдерге жататын бағдарламалау тілдері: C++, Java, Python.

Бағдарламада объекті түрінде көрсетілген деректерге айқын көңіл бөлінеді. Объектілер хабар алмасу механизмі көмегімен өзара бір-бірімен әрекеттеседі. Программисттің мәселесі болып өзара әрекеттесу барысында керек нәтижеге қол жеткізуге болатын объектілерді жасау.

Объектілі-бағытталған бағдарламалау құрылымдық бағдарламалаумен салыстырғанда өте ауқымды және қиындатылған есептерді шешуге арналған. Объектілі-бағытталған бағдарламалау негізінде мұрагерлік, полиморфизм және инкапсуляция деген түсініктер жатыр. Инкапсуляция бойынша объектінің аз қолданылатын бөліктері жасырынған. Объект қандай да бір команданы ала отырып қандай класқа жататынына байланысты қалай өңдеу керектігін өзі шешеді. Барлық объектілер класс экземпляры болады, олар бір-біріне қатысты әкелік-ұрпақтық рөлде бола алады.

Ұрпақтық класстар әкелік класстың қасиеттерін иемденеді. 100%-дық мұрагерлік қажет болмаған жағдайда ұрпақтық класстағы әкелік класстың тәсілдерін қайта анықтайтын полиморфизм жүзеге асырылады. Инкапсуляция бұл деректер мен кодты дұрыс қолданбаудан және сыртқы араласудан қорғайтын, және оларды байланыстратын механизм. Объектіге бағытталған бағдарламалауда деректер мен код бірге бірігуі мүмкін. Бұл кезде объект құрылады. Басқа сөзбен айтқанда, объект инкапсуляцияны қолдайды [2, 152 бет].

Компьютердің логикалық схемасы. Программалау, компиляциялау және программаның орындалу орталары

Программалау жүйесі компьютердің программалық қамсыздандыруының құрамына жатады және ол программаларды жасау және оларды жөндеу жұмыстарын автоматтандыруды қамтамасыз ететін құралдардың жиыны болып табылады. Программалау жүйесінің құрамы мына төмендегі схемада бейнеленген:

Программалау процесі үш кезеңге бөлінеді:

- есепті шешудің алгоритмін құру;
- программа құру;
- жасалған программаны тексеру.

Екінші кезеңдегі, яғни программа құрадағы қиындық адамның тек машина тілінде ғана программа жасауына байланысты болады. Компьютерді пайдаланудың алғашқы жылдарында әрқайсының әр түрлі компьютерлердің пайда болуы бұл қиындықты тереңдетіп жібереді. Сондықтан нәтижесінде бір компьютерге арнап жасаған программаны басқа компьютерге пайдалануға болмайтын болды. Бұл бір алгоритмнің өзін әр түрлі компьютерлерде орындау үшін әрқайсына жеке-жеке программа құруға мәжбүр етіп жұмысты қиындатып жіберді.

Машина тілінде программалау деп программаға енетін командалардың реальды кодтарын тікелей жазуды айтамыз.

Командалардың кодтары әр түрлі санақ жүйелерінде берілуі мүмкін:

- екілік;
- сегіздік;
- он алтылық;

Ассамблер тілінде программаның барлық элементтері таңбалармен берілетіндігі оның басты артықшылығы болып табылады. Басқаша айтқанда ассамблер тілінің машина командаларының цифрлық кодтарын әріптермен немесе әріп-цифрлармен таңбалауға және деректердің таңбалық аттарын пайдалануға мүмкіндік беретіндігі оның машина тілінен айырмашылығы болып табылады.

Машина тілінен өзгеше программалау тілінде құрылған программа компьютерде орындалу үшін жарамды түрде, яғни машина тіліне түрлендіруі тиіс. Мұндай түрлендіру *трансляциялау* деп аталады.

Жоғары деңгейлі тілдің трансляторы трансляциялау кезеңдерінің өту реттілігіне және программа операторының орындалуына байланысты *интерпретатор* немесе *компилятор* деп аталады. Интерпретатор әрбір жеке операторды трансляциялаған соң, оның тікелей тез орындалуын қамтамасыз етеді, яғни трансляция кезеңі мен орындалу кезеңі кезекпен қайталанады.

Компьютердің жадында тұрақты сақталатын стандартты бағыныңқы программалар жиыны стандартты бағыныңқы программалар кітапханасын құрады. Мұндай кітапхана құрамында бірнеше ондаған программалардан бірнеше жүздеген программаларға дейін болады.

Құрастырушы әрбір машина командасының және деректердің әр бір элементтің операциялық жүйедегі өз орнын анықтайды және модульдердің бірімен-бірінің қатынас жасауын қамтамасыз етеді.

Пайдаланушы құрған программадағы қателіктерді іздеуді және жөндеуді тездетіп қысқарту үшін программалау жүйесінің құрамына *жөндеуші* деп аталатын программа ендірілген [3, 222 бет].

Соңғы жылдары жоғары білім беру жүйесінде «Borland C++ Builder программалау тілі» атты пәніне ерекше көңіл бөле бастады бастады. Бірақ та, оны оқытудың мемлекеттік стандарты әлі жасала қойған жоқ. Әрі осы тіл туралы қазақша жазылған оқу- әдістемелік құралдары өте қанағатсыз көрінеді. Бұл мәселені арнайы зерттеп көру тұрғысынан аталмыш мақаланың тақырыбы таңдап алынған.

Borland C++ Builder программалау тілі жоғары деңгейлік күрделі тіл болғандықтан оның өзіндік ерелшелігі мен қиын жақтары оқыту әдістемесінде ойластырылуы керек. Мәселен, оқыту әдістемесінің мазмұны мен бағыт-бағдарын айқындаумен қатар жаңа идеялар мен технологияларды пайдалану жолдары қарастырылады.

Қазіргі кезде әлемдік білім кеңістігінен өз орнын алу үшін педогогикалық теорияны және оқыту процесін жетілдіру жұмыстары кең жүрілуде. Осыған

байланысты жоғары деңгейлік Borland C++ Builder программалау тілін оқытудың жаңа технологиясын пайдалану жолы болып табылады. Мақалада осы проблема теориялық тұрғыдан талданып, практикалық мысалдар арқылы көрсетіп түсіндіруге талпыныс жасалған. Зерттеу нәтижесі көрсеткендей, аталмыш Borland C++ Builder программалау тілін басынан дұрыстап оқыту үшін шешімін ойластыратын мынадай ғылыми-практикалық мәселелер тұр.

Пайдаланылған дереккөздер тізімі:

1. Концепция информатизации образования РК;
2. Ж.К.Нұрбекова, Қ.М.Байғушева, А.З.Даутова. Қазіргі бағдарламалау тілдері.
3. Бобровский С. Технология Петагона на службе российских программистов. Программная инженерия.-СПб.:Питер,200
4. Сайт [http://www.lib.asw1.ru/books/methodology/ programming](http://www.lib.asw1.ru/books/methodology/programming).

Айтбенова А.А.¹, Аманжол Б.М.²

1. *Ғылыми жетекші, педагогика магистрі, аға оқытушы*
2. *«Информатика» мамандығының 1 курс студенті, информатика және компьютерлік технологиялар кафедрасы*

DELPHI ОРТАСЫНДА «ҚАЗАҚ ТІЛІН ҮЙРЕНУ» ЭЛЕКТРОНДЫ ОҚУЛЫҒЫН ЖАСАУ ЖӘНЕ ҚОЛДАНУ

Ақпараттық технологиялар мен компьютерлік телекоммуникацияларды жыл сайын білім жүйесінде белсене қолданып келе жатыр. Сонымен қатар электронды оқулықтар арқылы сабақтарды түсіндіру де кең етек алуда. Электронды оқулықтарды тек қана сабақ барысында емес, өзін өзі үйрету барысында да қолдануға болады.

Электронды оқулық – бұл компьютерді қолданып, оқу курсы өз бетімен немесе оқытушының көмегімен меңгеруді қамтамасыз ететін программалық әдістемелік кешен. Электронды оқулық немесе курс үш компоненттен тұрады: курстың негізгі ақпараттық бөлігі мазмұндалған презентациялық құрамадан; алынған білімді бекітетін жаттығулардан; оқушының бағасын дұрыс бағалайтын тест жүйесінен. Оқыту программалары оқыту үрдісінде жетістікпен қолданылуда. Оқыту программасын жасау – өте күрделі және көп еңбектенуді қажет ететін жұмыс. Негізгі көңіл компьютерлік эффект немесе өнімдегі ерекше жаңа қабылдауларға бөлінеді. Жарнамалық жобалар мен компьютерлік басылымдар ең жиі қолданылатын компакт-дисктердің сипаттамасына қарау жеткілікті. Бірінші орында жазылған ақпараттың көлемі, иллюстрация, аудио және бейнеүзінділердің саны және олардың ұзақтығы. Ақпараттың сапасы жайлы ақпарат жоқ, ең жақсы жағдайда өнім басылымның электронды көшірмесі ретінде беріледі. Осылайша берілген оқулықтар, энциклопедиялар және басқа басылымдар оқу құралы ретінде қолданылуы мүмкін. Олар әр түрлі