

Отаханов Н. А.

DELPHI тилини ўқитишдаги айрим муаммолар ва уларни бартараф қилиш услубиёти



Отаханов Н. А.

DELPHI тилини ўқитишдаги айрим муаммолар ва уларни бартараф қилиш услубиёти



**ЎЗБЕКИСТОН РЕСПУБЛИКАСИ ОЛИЙ ВА ЎРТА
МАХСУС ТАЪЛИМ ВАЗИРЛИГИ**

НАМАНГАН ДАВЛАТ УНИВЕРСИТЕТИ

Отаханов Н. А.

**DELPHI тилини ўқитишдаги
айрим муаммолар ва уларни бартараф
қилиш услубиёти**

*НамДУ ўқув-услубий кенгашининг
2008 йил 3 июлдаги 10-сонли қарори билан
услубий қўлланма сифатида тавсия этилган*

«ФАРҒОНА» нашриёти
2008 йил

**Н. А. Отаханов. DELPHI тилини ўқитишдаги
айрим муаммолар ва уларни бартараф қилиш услубиёти**

«ФАРҒОНА» нашриёти, 2008 йил – 32 б.

Ушбу услубий қўлланма DELPHI объектга йўналтирилган дастурлаш тилини ўқитиш соҳасидаги айрим муаммолар ва уларни бартараф қилиш усуллари очиқ беришга қаратилган.

Китобча аспирантлар, дастурлаш тиллари бўйича ўқитувчиларга, услубчилар, олий ўқув юртлари, лицейлар, касб-хунар коллежлари талабаларига мўлжалланган.

Такризчилар:

1. НамДУ “Амалий математика ва АТ” кафедрасининг доценти,
ф.м.ф.н. А. Имомов
2. НамМПИ “Информатика ва АТ” кафедрасининг
доценти, ф.м.ф.н. С. Ирискулов

Кириш

Объектга йўналтирилган дастурлаш технологиялари – дастурий таъминотнинг инқирозига жавоб сифатида юзага келган дастурлаш технологиялари ҳисобланади. Бу инқирознинг сабаби шунда эдики, структурали дастурлаш методлари мураккаблик даражаси борган сари ортиб бораётган масалалар учун дастурий таъминот яратиш имконини бера олмай қолди. Бунинг натижасида турли лойиҳаларни бажариш режалари бузилди, қилинаётган ҳаражатлар белгиланган бюджетдан ортиб кетди, дастурий таъминотнинг функционаллиги бузилди, ҳатоликлари ортди.

Дастурий таъминотнинг энг муҳим томонларидан бири – унинг мураккаблик даражасидир. Бирор дастурчи системанинг барча хусусиятларини тўлиқ ҳисобга ола олмайди. Шунинг учун уни ишлаб чиқишда дастурчи ва бошқа мутахассисларнинг йирик жамоаси қатнашади. Демак, қўйилган масалага тўғридан-тўғри боғлиқ бўлган мураккабликларга ана шу жамоанинг ишини бир мақсадга қаратилган бошқариш ҳам қўшилади. Анъанавий дастурлаш тилларида бундай мураккабликларни ҳал қилишда “ажрат ва бошқар” принциpidан фойдаланилган. Яъни, масала кичик-кичик масалаларга ажратиб, кейин ҳар бир масала учун алоҳида дастур ишлаб чиқилган ва бирлаштирилган. Объектга йўналтирилган дастурлаш технологиялари эса масалага бошқача усулда ёндошади. Унда масалаларнинг ечиш учун керак бўладиган элементларни муаммоли соҳанинг турли абстракцияларига таалукли эканлиги асосий ўринда туради. Бу абстракциялар дастурчилар ишлаб чиқишган. Дастурчилар томонидан маълум бир соҳа ўрганилиб, унинг алоҳида объектлари ажратиб олинган. Бу объектлар учун масалаларни ечишда қўллаш мумкин бўлган хусусиятлар аниқланган. Эҳтиёжга қараб ҳар бир хусусият устида бажариш мумкин бўлган амаллар аниқланган. Сўнгра ўрганилаётган соҳанинг ҳар бир реал объектига мос дастурий объект ишлаб чиқилган.

Маълумки, компьютер ёрдамида ҳал қилинадиган ҳар бир масала учун махсус дастур ишлаб чиқиш ёзиш талаб қилинади. Бундай масалалар

синфининг кенгайиб бориши албатта янги-янги дастурлар яратишга олиб келади. Янги дастурларни яратиш учун «эски» дастурлаш тилларининг имконияти етмай қолганда ёки дастур яратиш жараёнини мукамаллаштириш учун янги дастурлаш тилига эҳтиёж пайдо бўлади. Бу масалани албатта малакали ва тажрибали дадстурчилар ёки дастурчи гуруҳлар томонидан ҳал қилинади. Ана шу дастурчиларни етиштириш бугунги кунда Республикамиз Олий Ўқув Юртларида талабаларга информатика ва информацион технологиялар каби фан ўқитувчиларининг энг асосий масаласи ҳисобланди.

Ушбу услубий қўлланма муаллифнинг Наманган давлат университетида “амалий математика ва информатика” мутахассислиги талаблари учун бир неча йил мобайнида ташкил қилган назарий ва амалий машғулотлари асосида шаклланган. Унда олий ўқув юртларининг талабаларига энг янги объектга йўналтирилган дастурлаш технологияларидан бири бўлган DELPHI дастурлаш тилини ўқитишдаги айрим муаммолар ва уларни ҳал қилиш йўл-йўриқлари келтирилган.

§-1. Дастурлашнинг янги технологиялари сифатида объектга йўналтирилган дастурлаш технологияларидан фойдаланиш

XX асрнинг 80 - йилларидан олдин ишлаб чиқилган компьютерлар учун катта дастурий системаларни ишлаб чиқиш жуда ҳам мушкул вазифа эди. Бунинг энг катта сабаби шу даврга хос бўлган компьютерларнинг имониятларининг чегараланганидадир. Дастурий комплексларни ишлаб чиқишда асосий чекланишлар компьютер тезкор хотирасининг сиғими, маълумотларни иккиламчи хотира қурилмаларидан (магнит ленталар, барабанлар ва х.к.) ўқиш тезлиги, процессорнинг ишлаш тезлиги (уларнинг такт частоталари бир неча юз микросекунд бўлган) билан боғлиқ. Бу даврдаги компьютерлар халқ ҳўжалигининг ҳисоб-китоб билан боғлиқ бўлган масалаларини ечиш учун мўлжалланган эди. Компьютерларда бошқа характердаги масалаларнинг ечишнинг иложи йўқ эди. Дастурчилар учун шу компьютерларда ечилаётган масала учун берилган маълумотларни ҳисобга олган ҳолда масаланинг ечиш алгоритмини ишлаб чиқиш биринчи ўриндаги вазифа ҳисобланган. Бу борада Н. Виртнинг машҳур *маълумотлар+алгоритм=дастур*¹ формуласини ёдга олиш етарли.

80-йиллардан кейин ишлаб чиқилган компьютерларнинг имкониятлари кенгайиб, ишлаб чиқариш нархи кескин пасайгани туфайли

¹ Н. Вирт. 1989. Алгоритмы и структуры данных. Москва, Мир, 360 стр.

улардан фойдаланиш самараси ортиб борди. Натижада, дастлаб матн ва графиклар учун муҳаррирлари ишлаб чиқилди, кейинчалик мультимедиа курилмаси ёрдамида товушли маълумотларни қайта ишлаш имкониятлари ҳам пайдо бўлди. Компьютер тасвирий имкониятларининг кенгайиши фойдаланувчилар учун қулай бўлган график мулоқот интерфейсини яратилишига сабаб бўлди. Натижада юқори мураккабликдаги катта амалий дастурларни яратиш фойдали ҳамда зарур бўлиб қолди. Дастурий воситаларни ишлаб чиқиш учун курул сифатида юқори босқичли алгоритмик тиллардан фойдаланилди. Бу дастурий воситалар дастурчи ва дастурчилар гуруҳи имкониятларини кенгайтириб, дастурий маҳсулотларнинг мураккаблик даражасининг ортишига ҳам сабаб бўлди.

70-80 йилларда дастурий маҳсулотларнинг мураккаблик даражасини ҳал қилиш учун кўплаб методлар ишлаб чиқилди. Уларнинг ичида юқоридан пастга усулидаги структурали дастурлаш кенг тарқалди. Бу усул юқори босқичли FORTRAN ва COBOL тилларининг топологиясига асосланган эди. Бу тилларда асосий базавий бирлик бўлиб қисм дастур хизмат қилади. Дастур шакли жихатидан дарахтни эслатиб, битта қисм дастур бошқасига мурожаат қилар эди. Структурали дастурлашда айнан шундай ёндошувдан фойдаланилади: алгоритмик декомпозиция катта масалаларни кичик масалачаларга бўлиш учун қўлланади.

XX асрнинг 60-70 йилларида ҳалқ хўжалигининг кўплаб масалаларини компьютер ёрдамида ечиш учун BASIC, PASKAL, FORTRAN каби дастурлаш тиллари, DBASE, CLIPPER, FOXBASE, KARAT каби маълумотлар базасини бошқарув тизимлари етарли бўлган. Бу тил ва тизимлар асосан IBM*86 типигаги компьютерларга мўлжалланган бўлиб, уларнинг асосий камчиликлари шу даврга ҳос бўлган компьютерларнинг имкониятлари билан боғлиқ эди:

- Фойдаланувчи учун қулай бўлган график интерфейс яратиш имкониятининг йўқлиги ёки жуда ҳам катта меҳнат талаб қилиши;
- Битта дастурда фойдаланиш мумкин бўлган ўзгарувчилар сонининг чегараланганлиги;
- Дастурда фойдаланиш мумкин бўлган сонли маълумотлар дивизионининг чегараланганлиги;
- Киритилаётган ва чиқарилаётган маълумотларни назорат қилиш;
- Буйруқлар тизимининг «камбағаллиги»;
- Графиклар билан ишлаш имкониятининг чегараланганлиги;
- Товушли маълумотларни қайта ишлашдаги камчиликлари;
- Маълумотлар базаси ва файллар билан ишлашдаги имкониятларнинг

чегараланганлиги;

- Янги типдаги маълумотлар фақат шартли равишда яратиш мумкин мумкинлиги ва ҳ.к.

Кейинги йилларда структурали дастурлашнинг ўндан ортиқ усуллари ишлаб чиқилди. Бу усулларнинг кўпчилиги битта мавзунинг турли кўринишларидан иборат бўлди. Аммо, Соммервил² уларни учта гуруҳга бўлишни таклиф қилди:

- Юқоридан қуйига қараб структурали лойиҳалаш усули;
- Маълумотлар оқимини ташкил қилиш усули;
- Объектга йўналтирилган лойиҳалаш.

Структурали лойиҳалаш усулига намуналар Йордан ва Константин³ асарларида келтирилган. Асосий ғоялар эса Виртнинг⁴ кўплаб илмий ишларида очиб берилган. Бу асарларда структурали дастурлашга нисбатан бўлган ҳар бир ёндошувда алгоритмик декомпозиция иштирок этади. Ёзилган дастурларнинг каттагина қисмининг асосини структурали дастурлаш принципи ташкил қилади. Структурали дастурлаш абстракцияларни ажратишга имкон бермайди, маълумотларни самарали ҳимоя қилишни таъминламайди, параллелизмни ташкил қилиш учун етарлича воситаларни таклиф қила олмайди. Структурали дастурлаш мураккаб системаларни ишлаб чиқишга имкон бермайди, объектга йўналтирилган дастурлашда самара бермайди.

Маълумотлар оқимини ташкил қилиш усули Джексон⁵ ҳамда Оррларнинг⁶ ишларида кенг ёритилган. Бу усулда дастурий системанинг структураси худди кирувчи маълумотларни чиқувчи маълумотлар қимига алмаштириш ташкил қилиш каби қурилади. Маълумотлар оқимини ташкил қилиш усули ҳам структурали дастурлаш усули каби бир қатор мураккаб масалаларда, хусусан ахборот таъминот системаларида қўлланилган. Бу масалаларда киритилётган ва чиқарилаётган маълумотлар ўртасида бевосита боғлиқлик мавжуд ҳамда дастурларнинг бажариш тезлигига катта эътибор берилмайди.

80-йилларга келиб, имкониятлари янада катта бўлган компьютерлар

² Sommerville.1. 1985. Sovftware Engineering. 2nd ed. Wotkthingam England: Addison Wesley , p.68.

³ Yordon E and Konstantine L. 1979. Structured Design. Englewood Cliffs. Nj:Prentice-Hall.

⁴ Virth N1986. A1gorithm and Data Structures. Englewood Cliffs. Nj:Prentice-Hall.

⁵ Jakson M. 1983. System Development. Englewood Cliffs. Nj:Prentice-Hall.

⁶ Orr K. 1971. Structured System development. New York. NY: Yordon Press.

ишлаб чиқилди. Структурали дастурлашнинг аҳамияти эса ўзгармади. Аммо Стайн⁷ шуни таъкидлайдики «дастурнинг ҳажми 100 000 сатрдан ўтганидан кейин структурали дастурлаш механизми ишламай қолар экан». Демак, катта дастурий системаларни ишлаб чиқишда структурали дастурлаш механизми ҳам ўз аҳамиятини йўқотади. Шу муносабат билан дастурий таъминотда объектга йўналтирилган дастурлаш технологияларига ўтиш таклиф қилинди.

Объектга йўналтирилган дастурлаш атамаси дастлаб 60-йилларнинг ўрталарида, Симула-67 дастурлаш тилида пайдо бўлган. Аммо, бу тил FORTRAN, ALGOL, PL/1 каби тилларга нисбатан ўзининг имкониятлари паст бўлганлиги ҳамда шу даврда ечиш талаб қилинган масалалар учун етарли воситаларни таклиф қила олмагани учун дастурчилар орасида кенг оммалаша олмади. Кейинчалик объектга йўналтирилган дастурлаш Smalltalk, Object Pascal, C++, Ada, ҳозирда эса C# каби юқори даражали дастурлаш тилларининг топологияси асосида ривожланди ва оммалашмоқда.

Объектга йўналтирилган дастурлаш – бу дастурлашнинг шундай янги йўналишики, дастурий система ўзаро алоқада бўлган объектлар мажмуаси сифатида қаралади ва ҳар бир объектни маълум бир классга мансуб ҳамда ҳар бир класс қандайдир шажарани ҳосил қилади деб ҳисобланади. Алоҳида олинган класс маълумотлар тўплами ва улар устида бажариладиган амалларнинг тўплами сифатида қаралади. Бу класснинг элементларига фақат шу классда аниқланган амаллар орқали мурожаат қилиш мумкин. Дастурдаги маълумотлар ва улар устида бажариладиган амаллар ўртасидаги ўзаро боғлиқлик анъанавий дастурлаш тилларига нисбатан дастурий системаларнинг ишончлилигини таъминлайди. Объектга йўналтирилган дастурлашнинг энг асосий тушунчаси объект ва класс ҳисобланади.

Объект. Олдимизда турган олма мевасини дастурлаш терминологияси орқали кўрайлик. Анъанавий дастурлашда биз уни қисмларга бўлиб ўрганганмиз: S-олма пўстининг юзи, J – олма мевасидаги шарбат ҳажми, F – пўст орасидаги мева оғирлиги, D-уруғлар сони ва х.к. Энди ана шу олмага рассом кўзи билан қарайлик. Олманинг расми бу мева эмас, балки олма мевасининг ясси текисликликдаги тасвиридир. Уни ҳар бири алоҳида ва бир-бирига боғлиқ бўлмаган бирор маълумотлар сегментида турган бир нечта маълумотлардан иборат бўлган кўринишда абстракциялаш мумкин

⁷ Stein J. March 1988 . Object-Oriented Programming and Database Design. Dr. Doobs journal of Software Tools for the Professional Programmier. № 137, p.18.

эмас. Олманинг компоненталари доимо биргаликда ва бу компоненталар орасидаги ўзаро алоқаларни сақлаган ҳолда қаралади. **Объект** - биз яшаётган оламдаги бирор элементга ҳос бўлган барча маълумот ва хулқларни, яъни шу элемент устида бажариш мумкин бўлган ҳаракатларни ифодалайди ҳамда маълумотларнинг тугал абстракциясидан иборат бўлади. Бу маълумот ва хулқлар объектга йўналтирилган дастурлаш атамашунослигида мос равишда **хусусият** ва **метод** деб аталади. Хусусиятни объектнинг **майдони** деб ҳам юритилади. Масалан, шашка объекти ранг, вертикал майдондаги ўрни, горизонтал майдондаги ўрни каби майдонларга, суриш, уриш, «дамка» га чиқиш, шашка тахтасидан четга олиш каби методларга эга бўлади. Майдон ва методлар биргаликда объектнинг аъзолари⁸ дейилади. Объектларнинг структураси уларнинг ўзаро алоқасини ифодалайди.

Эслатма: Объектлар ўзларининг барча характеристика ва хулқларининг ўзига ҳос томонларини биргаликда сақлайди.

Класс. Ҳар бир объект қандайдир битта классга таалукли бўлади. Класс – бу мураккаб структура бўлиб, ўз ичига маълумотларни, процедура ва функцияларни ифодалашдан ташқари, классларнинг вакили бўлмиш объектлар устида бажарилиши мумкин бўлган амалларни ҳам олади. Классдаги маълумотлар майдонлар, процедура ва функциялар эса методлар деб аталади. Класснинг структураси тушунчаси ҳам муҳим ҳисобланади. У система ичидаги воситалар бойлигини намоён қилади. Маълумки, бирор япроқдаги фотосинтез жараёнини ўрганиш учун шу япроқдаги битта хужайрани кўриш етарли, чунки, қолган хужайралар ҳам ўзини ана шгу ўрганилган хужайра каби тутати. Биз ҳам бирор класс, тип ёки бўлимга мансуб бўлган объектни кўрар эканмиз, фараз қилишимиз мумкинки, унинг хулқи ҳам шу типдаги бошқа объектларники каби бўлади.

Объектга йўналтирилган дастурлаш бизнинг хулқимизга ўхшаб кетадиган дастурлаш усулини ифодалайди. У дастурлаш тилларини ишлаб чиқишдаги янгиликларнинг табиий эволюцияси ҳисобланади. У барча аввалги дастурлаш тилларига қараганда структуралироқ, маълумотларни абстрактулашга уринишда абстрактулироқ ва модуллироқдир.

Объектга йўналтирилган дастурлаш ўзининг учта асосий хусусиятлари билан характерланади:

1. **Инкапсуляция** – ёзувларни шу ёзувларнинг майдонлари устида

⁸ Тугубалин Д. К. Объекто-ориентированное программирование в среде Delphi: Учеб. пособие. – Томск: Изд-во Том. гос. ун-та систем управл. и радиоэлектрон., 2004. – 112 с.

амаллар бажариш учун мўлжалланган функциялар ва процедуралар билан биргаликда олиб боради. Инкапсуляция яшириш принципига ҳам эга бўлади. Бунинг маъноси шуки, объект устида бажариш мумкин бўлган ҳаракатларни амалга ошириш воситалари шу объектдан фойдаланаётган дастурчилар кўзидан яшириб қўйилади. Дастурчи фақат шу объектнинг айрим метод ва майдонлари билангина ишлай олади. Бошқача айтганда, барча майдон ва методлар ички ва ташқи гуруҳларга бўлинади. Объектнинг ички аъзолари дастурчига «кўринмайди» ва объектнинг ҳулқи ва имкониятларини аниқлайди, ташқи аъзолари эса дастурчига «кўриниб туради» ҳамда объектни бошқариш имконини беради. Объектнинг дастурчига кўриниб, объектни бошқаришга ёрдам берадиган метод ва майдонлари (хусусиятлари) объектнинг интерфейси деб аталади. Дастурчи объект билан ишлаш учун унинг интерфейсини билиши кифоя. Масалан, енгил автомобилни бошқаришни ўрганиш учун унинг моторининг ишлаш принципи, ғилдиракларнинг бурилишини, тормоз механизмини ўрганишнинг ҳожати йўқ, рулни бураш, педал ёки узатмалар кутиси ричагини босишни билиш етарли.

2. **Ворислик** – янги объектни аниқлашга эҳтиёж пайдо бўлганда олдиндан маълум бўлган объектлардан фойдаланиш. Объектлар ўзларини яратишда иштирок этган ота объектларнинг характеристика ва ҳулқларини мерос қилиб олишлари мумкин. ОЙД концепцияси янги классларни мавжуд классларга янги майдонлар, хусусиятлар ва методларни қўшиш орқали яратиш имконини ҳам беради. Янги классларни ташкил қилишнинг бундай усули юзага келтириш деб аталади. Бу ҳолда юзага келган янги класс ўзининг базавий ота классига ҳос бўлган хусусият ва методларини мерос олади. Ҳаётдан мисол қилиб, ҳашаротлар классини олиш мумкин. У икки гуруҳга бўлинади: қанотлилар ва қанотсизлар. Қанотли ҳашаротларга капалаклар, пашшалари, парвоналар ва х.к. лар киради. Шунинг учун, пашшаларни қайтадан қанотли деб таърифлашнинг ҳожати йўқ, у бу хусусиятни ота классидан бўлган қанотли ҳашаротлар классидан мерос қилиб олади.

3. **Полиморфизм** – ҳаракатга бирор ном бериш ҳамда ундан биргаликда объектлар шажарасининг қуйи ва юқори қисмларида фойдаланиш. Бунда шажаранинг ҳар бир объекти бу ҳаракатни ўзи учун ҳос бўлган усул билан бажаради. Полиморфизм — бу турли классларга кирган методлар учун бир хил номлардан фойдаланиш имкониятидир. Полиморфизм концепциясида объектга нисбатан метод қўлланганида

айнан объектнинг классига мос келувчи методдан фойдаланишни таъминлайди.

Object Pascal дастурлаш тили объектга йўналтирилган дастурлашга ҳос бўлган барча воситаларни бера олади: структуралилик, модуллиқ, катта абстрактлилик. Бу характеристикаларнинг ҳаммаси хизмат кўрсатиш учун жуда ҳам содда, осонгина бошқа вазиятларга мослаштирила оладиган, кучлироқ структурага эга бўлган кодларда ўз аксини топади.

Объектга йўналтирилган дастурлаш узоқ йиллар давомида анъанавий, яъни стандарт ҳисобланган дастурлашга нисбатан ҳос бўлган тасаввурларни бир четга қўйишни талаб қилади. Натижада объектга йўналтирилган дастурлаш жуда ҳам содда, кўргазмалилиги юқори бўлиб, дастурий таъминот яратишдаги кўплаб муаммоларни ҳал қилишнинг жуда ажойиб воситасига айланади.

Юқорида айтилган фикрларни ҳисобга олсак, объектга йўналтирилган дастурлаш технологиялари қуйидагича масалаларни ҳал қилиши мумкин:

- анъанавий дастурлаш тилларида мавжуд бўлган камчиликларни бартараф қилиш;
- анъанавий дастурлаш тиллари ёрдамида ечиб бўлмайдиган ёки жуда катта қийинчиликлар билан ечилиши мумкин бўлган масалаларни ҳал қилиш;
- қайта ишлаш мумкин бўлган маълумотлар ва уларнинг типлари доираси анъанавий дастурлаш тилларига нисбатан анча кенг;
- фойдаланувчилар учун қулай бўлган мулоқот интерфейсини яратиш;
- киритилаётган ва чиқарилаётган турли типдаги маълумотларни назорат қилиш;
- янги типдаги маълумотлар, класслар ва модулларни осонгина ташкил этиш ва маълумотларни назорат қилиш;
- мультимедиа ва анимацион воситаларидан фойдаланиб, турли даражадаги товушли ва ҳаракатли эффектларни ҳосил қилиш ва қайта ишлаш;
- маълумотлар базаси ва ундаги маълумотлар устида амалларни бажариш, SQL сўровномалари ёрдамида маълумотларни қидириб топиш каби масалалар жуда осонлик билан ҳал қилиш;
- OLE контейнери ёрдамида WINDOWS муҳити учун мўлжалланган иловалардаги объеклар билан ишлаш;
- фойдаланувчилар учун яратилган дастурий таъминотдан фойдаланиш учун ёрдамчи маълумотномалар тизимини яратиш;

- дастурий таъминотни бошқа компьютерларга кўчириш учун ўрнатувчи дискларни яратиш;
- дастур матнини ташкил қилишда юзага келиши мумкин бўлган хатоликлар билан ишлаш масаласини ҳал қилиш ва ҳ.к.

Кўриниб турибдики, ечилаётган масалаларни анъанавий дастурлаш тиллари ёрдамида ечишда юзага келиши мумкин бўлган каттакон бўшлиқни объектга йўналтирилган дастурлаш технологиялари тўлдиради ҳамда замонавий дастурлашнинг кўплаб талабларига жавоб беради. Шундай экан, Олий ўқув юртларида анъанавий дастурлаш тилларидан воз кечиб, объектга йўналтирилган дастурлаш технологияларини ўқитишга ўтиш тавсия қилинади.

§-2. ОБЪЕКТЛИ АЛГОРИТМ

Биз яхши биламизки, алгоритм тушунчаси буюк ўзбек математиги Ал-Хоразмий номидан келиб чикиб, унинг машҳур «Ал-жабр ва ал-Муқобала» асарининг тарихий аҳамияти билан боғлиқ.

XX асрнинг охири ва XXI асрнинг бошларида дастурчилар ихтиёрига ўз имкониятларига кўра анъанавий дастурлаш тилларига қараганда бир неча марта бой ва қулай бўлган объектга йўналтирилган дастурлаш технологиялари таклиф қилинди. Бу технологияларнинг кейинги версиялари аввалгиларига қараганда мукамалроқ ҳамда борган сари ўз ичига кўплаб INTERNET тизими, тармоқлар, анимациялар ва маълумотлар базалари билан ишлаш каби янги имкониятлар учун зарур бўлган компоненталарни олиш ҳисобига кенгайиб бормоқда.

Бугунги кунга келиб, дастурлаш тилларини ўрганувчилар учун бир қараганда, алгоритм тушунчасини керак эмас, унинг дастурлаш билан алоқаси йўқ қабилидаги тасаввурлар пайдо бўлиб қолди. Улар масалаларни ечиш босқичлари, алгоритм қуришнинг йўл-йўриқлари билан танишмасдан туриб, дастурлаш соҳасига киришга ҳаракат қилишади. Натижада, улар дастурлаш тилларида кўзда тутилган қонун-қоидалар, буйруқлар тизимини етарли даражада ўрганишлари (шунчаки ёдлашлари) мумкин, аммо, аниқ қўйилган масала учун дастур ёзишда жуда катта қийинчиликларга дуч келишади. Ана шундай вазиятларга тушиб қолмаслик учун, биз бу дастурчиларга дастурлашни алгоритм қуришдан бошлашни тавсия қиламиз.

Бир қараганда, анъанавий дастурлаш тилларига ҳос бўлган алгоритм тушунчаси ўз аҳамиятини йўқотгандек бўлмоқда. Аслида ҳам шундайми? Бизнинг назаримизда бу саволга «йўқ» деб жавоб бериш мумкин. Чунки, ҳозирги кунда бу тушунча бошқача кўринишда, яъни замонавий дастурлаш тилларининг энг муҳим тушунчаси бўлган объектларга нисбатан

қўлланмоқда. Шунинг учун бу тушунчани янгича талқин қилишга тўғри келади.

Маълумки, ихтиёрий масалани ҳал қилиш учун махсус алгоритм ишлаб чиқиш талаб қилинади. Шундай алгоритмнинг мавжуд эмаслиги масала ечимининг мавжуд эмаслигини англатади.

Замонавий объектга йўналтирилган дастурлаш тиллари учун Н. Виртнинг «алгоритм+берилганлар=дастур» таърифи бошқачароқ талқин қилинади. Асосий эътиборни алгоритм тузишга эмас, балки маълумотларнинг берилганлар структурасига қаратилади.

Алгоритмларни ишлаб чиқишда объект, класс ва методлардан кенг фойдаланилади. Класс – бу мураккаб структура бўлиб, ўз ичига маълумотларни, процедура ва функцияларни ифодалашдан ташқари, классларнинг вакили бўлиши объектлар устида бажарилиши мумкин бўлган амалларни ҳам олади. Классдаги маълумотлар майдонлар, процедура ва функциялар эса методлар деб аталади.

Таъриф. Объектли алгоритм деб қўйилган масалани тўла ҳал қилиш учун зарур бўлган объектларнинг хусусиятларини ўзгартириш учун ижрочининг бажариши талаб қилинган амаллар ва методлар кетма-кетлигининг қатъий тартибига айтилади.

Объектли алгоритм қуриш учун унинг қандай хусусиятларга эга эканлиги ҳамда ана шу хусусиятлар устида бажариш мумкин бўлган амалларни билиш лозим. Хусусиятлар ва уларнинг кийматлари бир-биридан “.” (нуқта) белгиси ёрдамида ажратиб кўрсатилади.

1-мисол. Уйни иситгич ёрдамида иситиш талаб қилинган бўлсин. Бу масалани ечишда ижрочи, яъни уй эгаси иситгич объектидан фойдаланади. Иситгич объекти розетка, ўчиргич ҳамда ўзининг махсус иш режимлари каби хусусиятларга эга. Ижрочининг тармоқда электр кучланишининг мавжудлиги қизиқтирмайди. Чунки, кучланишни муҳит таъминлаб беради. Маълумки, розетка хусусияти умумий электр тармоғига «уланган» ёки «уланмаган», ўчиргич хусусияти «ёқилган» ёки «ўчирилган», иш режими хусусияти эса «кучли», «ўрта» ва «кучсиз» кийматларни олиши мумкин. Демак, ижрочи бу хусусиятларни ўзгартириб, олдида қўйилган масалани ҳал қилиши мумкин.

Бу мисолдаги хусусиятларни қуйидаги тартибда белгилаб қўйиш мумкин:

- 1) ўчиргич.уланган (ёки ўчиргич.рост);
- 2) Иш режими.кучли;
- 3) Розетка. ўчирилган (ёки Розетка.ёлғон).

2-мисол. Шашка донаси ранг, горизонтал ва вертикалдаги ўрни каби хусусиятларга ҳамда юриш, уриш ва тахтадан четга олиш каби амалларга (методларга) эга. Ўйинчи берилган хусусиятларни ўзгартириб, мумкин бўлган амалларни бажариб, ўйин тақдирини ҳал қилиши мумкин.

Классининг структураси тушунчаси ҳам муҳим ҳисобланади. У система ичидаги воситалар бойлигини намоён қилади. Маълумки, бирор япроқдаги фотосинтез жараёнини ўрганиш учун шу япроқдаги битта хужайрани кўриш етарли, чунки, қолган хужайралар ҳам ўзини ана шгу ўрганилган хужайра каби тутати. Биз ҳам алгоритм қуриш жараёнида бирор класс, тип ёки бўлимга мансуб бўлган объектни кўрар эканмиз, фараз қилишимиз мумкинки, унинг ҳулқи ҳам шу типдаги бошқа объектларники каби бўлади.

Объектли алгоритм қуйидаги талабларга жавоб бериши лозим, акс ҳолда у объектли алгоритм бўла олмайди.

1. Алгоритмда фойдаланилаётган ҳар бир объект қандайдир битта классга таалуқли бўлиши лозим.
2. Алгоритм фойдаланувчилар учун мўлжалланган, яъни масала шартида берилган маълумотларни киритиш ҳамда талаб қилинган маълумотларни чиқариш учун интерфейс, яъни мулоқот ойнасига эга бўлиши шарт;
3. Алгоритмнинг ҳар бир қадами буйруқ тарзида расмийлаштирилади.
4. Алгоритм чекли сондаги қадам ва буйруқлардан иборат бўлиши керак.
5. Амал ёки методларда фойдаланилаётган ўзгарувчиларнинг қийматлари олдиндан аниқланиши лозим;
6. Ҳар бир амал ёки методнинг натижаси бир қийматли бўлади;
7. Интерфейсга ишни бошлаш учун бир ёки бир нечта тугмали ойналар киритилиши ҳамда бу тугмаларнинг бирини танланишига қараб алгоритм реакциясини аниқлаш керак.
8. Алгоритм оммавий бўлиши лозим.

Бу талабларнинг айримларини изоҳлаймиз.

1-талабга кўра, ҳар бир объект бирор классга таалуқли бўлади, яъни шу классга ҳос бўлган барча хусусиятларни олган бўлади. Эҳтиёж пайдо бўлганда бу хусусиятларни алгоритм қадамлари ёрдамида ўзгартирилиши мумкин. Масалан, 1-мисолдаги шашка донаси ранги, тахтанинг горизонтал ва вертикал қаторидаги турган ўрни каби хусусиятларга эга. Ўйин жараёнида шашка донасининг у ёки бу хусусиятлари ижрочи, яъни ўйинчи томонидан ўзгартирилади.

2-талабга фойдаланувчи билан алгоритм ижрочиси ўртасидаги мулоқот ойнаси бўлган интерфейс яратилиши керак. Бу интерфейсдаги барча объектларнинг хусусиятларини ўзгартириш эҳтиёжга қараб ёки олдиндан, ёки алгоритм кўрсатмаларининг бажарилиши давомида ўзгартирилади.

4-талабда алгоритмнинг қадамлари сони чекли бўлиш таъкидланган. Демак, ижрочи чекли сондаги элементар амалларни бажарганидан сўнг, масаланинг ечимига эришиши лозим. Айрим ҳолларда бу талабни бузилиши натижасида «чексиз цикл» эффекти содир бўлади ва бу цикл ҳеч қачон тугамайди. Масалан, қуйидаги буйруқларга эътибор беринг:

1. Ҳисоблансин $x:=1$;
2. Ҳисоблансин $x:=x+1$;
3. 1-қадамга ўтилсин.

Демак, алгоритм курилатганда бундай вазиятларни албатта назарда тутиш лозим.

5-талаб бўйича шуни айтиш мумкинки, амал ёки методларда қатнашаётган ўзгарувчиларнинг қийматларини олдиндан аниқлаш масаланинг тўғри ҳал қилинишига олиб боради. Бундай ўзгарувчиларнинг бошланғич қийматларини кўрсатмаслик уларни ҳар қил талқин қилинишига олиб келади. Масалан, ижрочи қиймати олдиндан аниқланмаган x ўзгарувчи учун

$$y:=2*x-1;$$

буйруғини қай тарзда бажариши лозим?

6-ҳоссага кўра, алгоритмнинг қадамларини ташкил қилишда «бирор сонни ўйланг» ёки $x_{1,2}:=\pm\ldots$ қабилдаги кўринишидаги буйруқлардан фойдаланиш мумкин эмас.

8-талабда алгоритмнинг оммавий бўлиши таъкидланган. Бу шуни англатадики, биринчидан, объектли алгоритм мумкин бўлган барча бошланғич қийматларни қайта ишлашга мўлжалланган бўлади, иккинчидан, алгоритм қўйилган масаланинг умумий ҳоли учун курилади ҳамда уни шу масаланинг ихтиёрий хусусий ҳоли учун қўллаш мумкин бўлади, учинчидан, шу масала билан битта синфга кирган масалалар учун ҳам қўллаш мумкин бўлади.

Объектларнинг хусусиятлари икки ҳил усулда кўрсатилиши мумкин.

1. Алгоритмнинг хусусиятларини кўрсатмаларини бажаришдан олдиндан белгилаб қўйиш. (1-мисолга қаранг.) Кўриниб турибдики, уйни иситиш учун иситгичнинг хусусиятларини белгилаб қўйиш талаб қилинади.

2) Алгоритмнинг хусусиятларини бажариш вақтида ўзгартириш. Зарур бўлганда объектнинг у ёки бу хусусиятини «эски» қийматини «янги»

си билан алмаштириб қўйишга эҳтиёж пайдо бўлиши мумкин. Масалан, иситгич ишлаб турган вақтда уй ҳаддан ташқари исиб кетган бўлса, унинг иш режими қийматини

Иш Режими:=ўрта (ёки **Иш Режими:**=кучсиз) ; (*)

тарзида ўзгартириб қўйиш мумкин.

Кўпинча, берилган масалаларни ҳал қилиш учун уларнинг хусусиятларини тўғридан-тўғри ўзгартиришнинг иложи бўлмайди. Уларнинг қиймати қандайдир амалларнинг натижасига боғлиқ бўлиб қолади. Демак, объектли алгоритмларда қўшимча равишда қандайдир ўзгарувчиларнинг қийматларини киритилиши, ижрочи томонидан танланиши ёки кўрсатиладиган амаллар ёрдамида ҳисоблаб топилиши мумкин. Бундан ташқари, зарур бўлган айрим маълумотларнинг кўрсатилишини талаб қилиш мумкин. Бу мақсадлар учун объектли алгоритмларнинг махсус киритиш ва чиқариш ойналаридан эҳтиёжга қараб фойдаланиш мумкин. Бу ойналардаги маълумотлар матн шаклида бўлади. (*) буйруқни киритиш ойнасидан фойдаланиб,

1. Киритилсин(a);

2. Иш Режими:=a;

кўринишида ҳам ёзиш мумкин.

Зарур бўлганда, киритилган матн шаклидаги маълумотларни махсус функциялар ёрдамида сонли ёки бошқа типдаги маълумотларга айлантириш мумкин. Бунда мос равишда **сон(x)** ёки **матн(x)** функцияларидан фойдаланилади.

3-мисол. $ax^2+bx-c=0$ квадрат тенгламани ечинг.

Ечиш. Бу масалани ҳал қилиш учун **a**, **b** ва **c** ўзгарувчиларнинг қийматларини киритилиши лозим. Уларни киритиш учун матнли маълумотларга мўлжалланган **Киритилсин** ойнасидан фойдаланилади.

1. Киритилсин a.матн;

2. Киритилсин b. матн;

3. Киритилсин c. матн;

4. a1:=сон(a.матн);

5. b1:=сон(b.матн);

6. c1:=сон(c.матн);

7. D:=b²-4ac;

8. агар D<=0 бўлса чиқарилсин(“ечим йўқ”) ; 13 га ўт;

$$9. x_1 = \frac{-b + \sqrt{D}}{2a}$$

$$10. x_2 = \frac{-b - \sqrt{D}}{2a}$$

11. Чиқарилсин матн(x_1);

12. Чиқарилсин матн(x_2);

13. Тамом.

Айрим ҳолларда берилган масалани ечиш учун бажарилиши талаб қилинган амалларнинг сони жуда катта бўлиши мумкин. Турган гапки, бу ҳолда алгоритм қуриш жараёни мураккаблашиб кетади. Юзага келган ноқулайликнинг олдини олиш учун, алгоритмларни блокли усулда қуриш тавсия қилинади. Блокли усулда берилган масалани бир нечта кичик-кичик масалаларга (блокларга) ажратилади ва бу блокларни бир-бирига масаланинг характерида келиб чиққан ҳолда битта мантиқий кетма-кетликка бириктириш механизми ишлаб чиқилади. Ҳар бир блок учун алгоритм ёзилади ва шундай кейин бу блокларни ишлаб чиқилган кетма-кетлик асосида бириктирилади.

4-мисол. Учбурчак учларининг координаталари берилган бўлсин. Координаталар боши шу учбурчак ичида ётадими?

Берилган масала учун мантиқий кетма-кетлик қуйидагича қурилади:

1. Берилган учбурчакнинг юзи S бўлсин;

2. Координаталар боши бўлган $(0, 0)$ нукта учбурчакнинг учлари билан бириктирилсин;

3. Ҳосил бўлган учбурчакларнинг юзалари ҳисобланиб, мос равишда S_1 , S_2 ва S_3 билан белгилансин;

4. Агар $S = S_1 + S_2 + S_3$ бўлса чиқарилсин(“ҳа”) акс ҳолда чиқарилсин(“йўқ”);

5. Тамом.

Ташкил қилинган блокларнинг алгоритмларини қуриш жуда ҳам содда бўлгани учун, уларни келтирмаймиз.

Объектли алгоритмлардан энг замонавий дастурлаш тиллари бўлган объектга йўналтирилган дастурлаш тилларини ўқитишда кенг фойдаланиш мумкин.

§-3. Талабаларни DELPHI дастурлаш тилини ўрганишга тайёрлаш босқичлари ҳусусида

Биз яхши биламизки, компьютерларнинг янги авлодлари жуда ҳам тез пайдо бўлмоқда. 60-65 йиллик давр мобайнида компьютерларнинг тўртта авлоди яратилди, ҳозирги кунда келиб ҳам бу анъана давом этиб, янги компьютер авлодлари устида ишлар олиб борилмоқда. Ҳар бир авлод аввалгисидан ўзининг структураси билан, ишлаш принципи,

имкониятларининг бойлиги, амалларнинг бажариш тезлиги ва бошқа кўплаб техник жиҳатлари билан фарқ қилади. Ҳаттоки, бир авлод доирасида ишлаб чиқилган янги компьютерлар ҳам аввалгиларидан имкониятларининг кенглиги, пухталиги ҳамда қулайлиги билан ажралиб туради. Компьютерларнинг имкониятлари кенгайиб боргани сари, улар ёрдамида ҳал қилиш мумкин бўлган масалалар синфи ҳам катталашиб боради.

Бундан 15-20 йил аввал ҳалқ хўжалигининг кўплаб масалаларини компьютер ёрдамида ечиш учун BASIC, PASCAL, FORTRAN каби дастурлаш тиллари, DBASE, CLIPPER, FOXBASE, KARAT каби маълумотлар базасини бошқарув тизимлари етарли бўлган. Бу тил ва тизимлар асосан IBM286, IBM396 ва IBM486 типдаги компьютерларга мўлжалланган эди. Бу тилларнинг асосий камчиликлари шу даврга ҳос бўлган компьютерларнинг имкониятлари билан боғлиқ бўлиб, куйидагилардан иборат эди:

- Фойдаланувчи учун қулай бўлган график интерфейс яратиш имкониятининг йўқлиги ёки жуда ҳам катта меҳнат талаб қилиши;
- Янги типдаги маълумотлар фақат шартли равишда аниқланиши, масалан тўпламлар, элементлари саналадиган ва чегараланган типдаги маълумотларни эркин киритиш ва чиқаришнинг иложи йўқлиги;
- Киритилаётган ва чиқарилаётган маълумотларни назорат қилиш;
- Буйруқлар тизимининг «камбағаллиги»;
- Графиклар билан ишлаш имкониятининг чегараланганлиги;
- Товушли маълумотларни қайта ишлашдаги камчиликлари;
- Маълумотлар базаси ва файллар билан ишлашда имкониятларнинг чегараланганлиги ва ҳ.к.

PENTIUM типдаги компьютерларнинг дастлабки авлодлари пайдо бўлиши билан, уларнинг имкониятларини ҳисобга олган ҳолда юқорида айтиб ўтилган камчиликларни бартараф қилишга қаратилган янги дастурлаш тиллари яратила бошланди, эски дастурлаш тилларининг визуал вариантлари ишлаб чиқилди. Масалан, DELPHI, C++, VISUAL BASIC, VISUAL C++ каби тиллар ана шулар жумласидан ҳисобланади.

Ҳозирги кундаги ҳал қилиниши талаб қилинган кўплаб масалалар учун DELPHI дастурлаш тилининг имкониятлари етарли бўлиб, уни универсал дастурлаш тили деб аташ ҳам мумкин. Маълумки, DELPHI дастурлаш тили ёрдамида Windows операцион тизимида қўллаш учун ҳамда компьютернинг имкониятларидан янада тўлароқ фойдаланиш учун энг содда иловадан тортиб, то мураккаб маълумотлар базаси билан ишлаш учун иловалар, тармоқлар, шу жумладан интернет тармоғи билан ишлаш учун

ҳам дастурий таъминот яратиш мумкин. Бу ўринда шуни эсга олиш етарлики, DELPHI дастурлаш тили DELPHI дастурлаш тилида яратилган. Шунинг учун, Олий Ўқув Юртларида, нафақат компьютер билан боғлиқ мутахассисликлар учун, балки бошқа йўналиш талаблари учун ҳам DELPHI дастурлаш тилини ўрганиш мақсадга мувофиқ ҳисобланади.

Объектга йўналтирилган дастурлаш тиллари, шу жумладан DELPHI дастурлаш тили имкониятлари жиҳатидан, айниқса графика ва анимацион тизимлари жуда кенг ва бой тиллар ҳисоблангани учун, уни бирданига талабаларга ўргатиш анча мураккаб вазифа. Шунинг учун бу вазифани қадам-бақадам, маълум бир босқичлар асосида, «соддадан мураккабга» қараб олиб бориш тавсия қилинади. Биз бу жараёни қуйидаги босқичлардан иборат бўлишини таклиф қиламиз:

- талабаларни ЭХМ қурилмалари ва уларнинг ишлаш принциплари билан таништириш;
- масалаларнинг компьютерда ечиш босқичлари;
- алгоритм тушунчаси ва алгоритмларнинг ёзиш усулари;
- Object Pascal дастурлаш тили;
- объектга йўналтирилган дастурлаш технологиялари (DELPHI дастурлаш тили мисолида).

Биз юқорида таклиф қилган босқичларнинг назарий қисми Олий ва Ўрта Махсус Таълим Вазирлиги томонидан Олий Ўқув Юртларининг амалий математика ва АТ мутахассисликларига «дастурлаш асослари» фани бўйича 2006-2007 ўқув йиллари учун таклиф қилинган намунавий ўқув режасидан мустаҳкам ўрин олган. Амалий қисми эса шу мутахассислик учун «Компьютер амалиёти» фанининг намунавий дастурида ўз аксини топган.

I-босқичда белгиланган вазифани биз «Татбиқий математика ва информацион технологиялар фанига кириш» фани ёрдамида ҳал қилишимиз мумкин. Бу фан шартли икки қисмдан иборат бўлгани учун, унинг ишчи режасини тузишда аввал информацион технологиялар, сўнгра татбиқий математикага оид материалларни ўқитиш кўзда тутилиши керак. Акс ҳолда, талабаларга информацион технологияларнинг компьютер, унинг асосий қурилмалари, ишлаш принципи, санок системалари ва ахборотларни кодлашга оид бўлган маълумотларини «дастурлаш асослари» фанининг ишчи режасига киритишга тўғри келиб қолади. Бошқача айтганда, бир ҳил маълумотларни икки фан доирасида талабаларга ўргатишга тўғри келиб қолади. Бир ҳил маълумотни икки марта такрорлаш талабаларнинг зерикашига сабаб бўлиши мумкин. Агар эҳтиёж пайдо бўлса, бундай материаллар билан ишлаш учун «дастурлаш асослари» фанининг

ўқитувчиси фақат мустақил машғулотлардан фойдаланишини тавсия қиламиз.

II-IV- босқичлар юқорида таъкидлаганимиздек, «дастурлаш асослари» ва «компьютер амалиёти» фанларининг асосини ташкил этади.

Маълумки, Олий Ўқув Юртларининг «Амалий математика ва АТ» мутахассислигининг талабаларига дастлабки тўртта семестр давомида «Дастурлаш асослари» ҳамда «компьютер амалиёти» фанлари ўқитилади. I-семестрда асосий эътиборни масалаларни ЭХМ да ечишнинг босқичлари, масалаларни ечиш учун алгоритмлар қуришга қаратиш тавсия қилинади. Бунда талабалар алгоритмларнинг барча кўринишлари билан, яъни чизиқли, тармоқланувчи, такрорланувчи ҳамда ёрдамчи алгоритм тушунчалари билан чуқур таништирилади. Амалий машғулотларда эса шу мавзуларга доир масалалар ечиш бўйича мустаҳкам билим ва кўникмалар ҳосил қилинади. Бу мавзуларни талабаларга етказиш учун назарий машғулотларга 8 соат, амалий машғулотларга 10 соат, шунингдек «компьютер амалиёти» фани доирасидан лаборатория машғулотлари учун 10 соат вақт ажратиш етарли деб ўйлаймиз.

Гарчи ЎзМУ да 2006 йилда ишлаб чиқилган ва Олий ва Ўрта Махсус Таълим Вазирлиги томонидан таклиф этилган намунавий дастурига киритилмаган бўлса ҳам, қийматлар жадвали тушунчасини талабалар онгига етказишни тавсия қиламиз. Чунки, содда масалалар учун талаба қийматлар жадвалини тузар экан, ўзи келажақда ишлаб чиқиши мумкин бўлган дастурларни компьютер томонидан қандай тартибда бажарилиши ҳақида тасаввурга эга бўлади. Бу эса, шу талабалар ишлаб чиқадиغان дастурлар матнида мавжуд бўлиши мумкин бўлган турли характерга эга бўлган ҳатоликларни камайтиришга олиб келиши табиий. Шунинг учун, қийматлар жадвали тушунчасининг назарий асосларини маъруза соатларида талабаларга етказишларини ҳисобга олиб, ўқитувчилар ўз ишчи режаларига ҳеч бўлмаганда 2 соат амалий машғулот ажратишлари тавсия қилинади. Талабалар бу тушунча бўйича олган билимларини мустақил машғулотлар ҳисобига мустаҳкамлаб олишлари мумкин.

«Дастурлаш асослари» фанининг мантиқий давоми бўлган «Компьютер амалиёти» бўйича лаборатория ишларида ҳам қийматлар жадвали тушунчасини талабалар онгига чуқур етказиш мумкин. Шу фан доирасида ҳам, алгоритм тушунчаси билан боғлиқ дастлабки лаборатория ишларида, айниқса, талабалар ҳали бирор дастурлаш тилида дастур ёзишни ўрганганларича йўқ, аммо индивидуал топшириқлар асосида таклиф қилинган масалалар учун алгоритм ёзишни ўргатиш босқичида қийматлар жадвали тушунчасига талабалар диққатини жалб қилиш мумкин. Ёки

масалалар учун алгоритм ёзаётган талабадан масаланинг ечиш ғояси ва алгоритми, блок-схемасига қўшимча равишда бу алгоритмларнинг қийматлар жадвалини ҳам тўлдиришни талаб қилиш мумкин.

I-семестр учун ажратилган соатларнинг қолган қисмини Object Pascal дастурлаш тилига бағишлаш мумкин. Маълумки, DELPHI дастурлаш тили Object Pascal тилининг устига қурилган, бошқача айтганда DELPHI дастурлаш тилининг муҳити Турбо Паскал дастурлаш тилининг кенгайтирилган варианты бўлган Object Pascal дастурлаш тилидан фойдаланади. Бу икки тилнинг полиморфизм, инкапсуляция, ворислик каби ўхшаш томонлари жуда ҳам кўп. Бундан ташқари, қайта ишланадиган DELPHI дастурлаш тилидаги маълумотларнинг типлари ҳам Object Pascal тилидаги маълумотларнинг мантиқий кенгайтирилгани деб қараш мумкин. Шунинг учун ҳам, Олий Ўқув Юртлари талабаларига DELPHI дастурлаш тили ўргатишдан аввал, бошланғич босқич сифатида Object Pascal тилини ўргатиш тавсия қилинади. Бу тилни ўргатиш асосан, олий ўқув юртларида I ва II семестрларда амалга оширилади. Биз «программалаш асослари» фани бўйича ўз ишчи дастуримизни тузишда Олий ва Ўрта Махсус Таълим Вазирлигининг тавсияларини асос сифатида қабул қилиб, қўшимча равишда фаннинг специфик хусусиятларини ҳисобга олган ҳолда ташкил қилсак, мақсадга мувофиқ бўлади.

Бизнинг ишчи дастуримизда «Программалаш асослари» фанининг намунавий дастурида тилга олинмаган икки мавзуни киритишга қарор қилдик. Бу мавзулар «дастурларни ёзиш усуллари» ҳамда «Дастурларни оптимallasштириш» мавзуларидир. Дастурчилар учун бу икки мавзунинг ўта долзарблиги ва муҳимлигини изоҳлашнинг ҳождати йўқ.

Кўп йиллик тажрибаларимиз шуни кўрсатдики, «дастурлаш асослари» фани бўйича амалий машғулот соатлари етарли эмас. Ҳар бир семестрда 40 соатдан амалий машғулот режалаштирилган ҳалос. Кўриб чиқиладиган назарий мавзулар доираси эса анчагина кенг. Шунинг учун, ҳам бир мавзу бўйича 2-4 соатдан ортиқ амалий машғулот ўтказишнинг иложи йўқ. Ҳар бир 2 соатлик жуфтлик амалий машғулот мобайнида ўртача 4-6 та масалани кўриш мумкин. 4 соатлик амалий машғулотлар учун бу миқдор 8-10 тани ташкил қилади. Маълумки, дастурлаш соҳасида иккита бир хил масала деярли учрамайди ҳамда ҳар бир масалага алоҳида ёндошиш талаб қилинади. Буни ҳисобга олинса, юқорида айтилган 8-10 масала жуда ҳам кам. Масалаларни кўроқ ечиш учун мустақил машғулотлардан фойдаланиш ҳам кўзланган самарани бермаяпти. Бунинг асосий сабаби, биринчидан, дастурлашга доир ўзбек тилидаги дидактик материаллар кам, иккинчидан, ёзилган дастурларнинг компьютерларда текширишни ҳар доим

ҳам имконияти мавжуд эмас, ўқитувчилар томонидан намуна тарзида кўрсатиладиган масалалар ҳам кўп эмас. Шунинг учун биз II семестрнинг якуний қисмида бир нечта машғулотларни мавзуни ҳаётдан олинган турли мавзуларга доир бўлган типик масалаларга бағишламоқдамиз. Натижада талабалар алгоритм куриш ва Object Pascal дастурлаш тилининг буйруқлар тизими бўйича олган назарий ва амалий билимларини янада чуқурлаштириб, дастурлаш бўйича кўникмаларини малака даражасига кўтариб, дастурлаш соҳасида олган барча назарий билимларини амалиётга татбиқ қилишни, ҳаётий ва илмий фаолиятларида учраб турадиган масалалар учун дастурлаш ёзишни пухта ўзлаштиришмоқдалар.

Биз ана шу босқичлардан ўтган талабани объектга йўналтирилган дастурлаш технологияларини ўрганишга тайёр деб ҳисоблашимиз мумкин. Дастур ёзишнинг турли усуллари билан таниш, уларни оптималлаштириш йўлида етарлича тасаввурга эга, илмий-ҳаётий кўплаб масалаларни дастурлаш бўйича етарли кўникма ва малакага эга бўлган талабаларга энди бемалол Паскал тилига нисбатан кучлироқ, имкониятлари кенгроқ, катта-катта системаларни дастурлашга мўлжалланган энг замонавий объектга мўлжалланган дастурлаш тилларидан бирини, масалан, DELPHI дастурлаш тилини таклиф қилиш мумкин. Айтиш мумкинки, бу жараён талабалар учун ҳам қизиқ туюлади. Биринчидан, Паскал дастурлаш тилида бир неча ўнлаб буйруқлар ёрдамида ҳал қилинадиган масалани DELPHI тилидаги биттагина кўрсатма орқали ҳал қилиш мумкин. Масалан, паскал тилида файлдан бирор маълумотни қидириб топиш учун, бир нечта буйруқларни ёзиш талаб қилинса, объектга йўналтирилган дастурлаш тилида бу масалани битта сатрга жой бўладиган SQL-сўровномаси ёрдамида ҳал қилиш мумкин. Иккинчидан, Паскал тилида дастур ёзилганда асосий вақт ва эътибор уларнинг матнини компьютер хотирасига киритиш ва дастурни отладка қилишга сарфланади, объектга йўналтирилган дастурлашда эса бу жараёни тайёр шаблон ва эслатмалардан фойдаланиш, зарур объектларнинг хусусиятларини ўзгартириш эвазига енгиллаштириш мумкин.

Педагогиканинг энг асосий «соддадан мураккабга» принципи остида ташкил қилиб, мураккаб ва қизиқарли объектга йўналтирилган дастурлаш тилларини ўргатиш жараёнини содда Паскал дастурлаш тилини ўргатишдан бошлаш бизнинг бир неча йиллардан олиб бораётган тажрибаларимизда яхши натижа бериб қолган эди. Аммо, унинг ўрнига PASCAL тилининг ривожлантирилган OBJECT PASCAL варианты DEPLHI дастурлаш тилини ўрганишда яхшигина кўприк бўлишини назарда тутсак, I ва II семестрларда имкониятга қараб, OBJECT PASCAL дастурлаш тилини ўрганган маъқул.

§-4. Объектга йўналтирилган дастурлаш технологияларини ўқитишда турли типдаги маълумотлар билан ишлаш услубиёти.

Биз ушбу мавзуни муҳокама қилишдан аввал DELPHI дастурлаш тилида ишлаш мумкин бўлган маълумотларнинг типларини эслайлик.

Бутун тип. DELPHI тили еттита типдаги бутун сонли маълумотларни қабул қила олади: Shortint, Smailint, Longint, Int64, Byte, Word ҳамда Longword.

Бутун типдаги сонлар

1-жадвал

типи	Диапазони	ўлчами
Shortint	-128 ... 127	8 бит
Smallint	-32 768 ... 32 767	16 бит
Longint	-2 147 483 648... 2 147 483 647	32 бит
Int64	$-2^{63} \dots 2^{63} - 1$	64 бит
Byte	0...255	8 бит, ишорасиз
Word	0...65 535	16 бит, ишорасиз
Longword	0 ... 4 294 967 295	32 бит, ишорасиз

DELPHI тили энг универсал бутун типли маълумот ҳисобланадиган ва longint типига эквивалент бўлган Integer типдаги маълумотларни ҳам қабул қилади.

Ҳақиқий тип. DELPHI тилида олтита ҳақиқий типдаги маълумотлар мавжуд: real48, single, double, extended, comp, currency. Бу типлар бири-биридан қабул қиладиган қийматларининг диапазони, ишончли рақамларининг сони ва компьютер хотирасидан эгаллайдиган ҳажмлари билан фарқланади. 1-жадвалда бу типлар улар ораидаги фарқ келтирилган.

DELPHI тили Double типига эквивалент бўлган универсал ҳақиқий тип - Real типдаги маълумотларни ҳам қабул қилади.

Ҳақиқий типдаги маълумотлар

Жадвал-2

	Диапазон	Ишончли рақамлари	байт
Real48	$2.9 \times 10^{-39} \dots 1.7 \times 10^{38}$	11-12	06
Single	$1.5 \times 10^{-45} \dots 3.4 \times 10^{38}$	7-8	04
Double	$5.0 \times 10^{-324} \dots 1.7 \times 10^{308}$	15-16	08
Extended	$3.6 \times 10^{-4951} \dots 1.1 \times 10^{4932}$	19-20	10
Comp	$2^{-63}+1 \dots 2^{63}-1$	19-20	08

Currency	-922 337 203 685 477.5808 ... 922 337 203 685 477.5807	19-20	08
----------	---	-------	----

Турли типдаги маълумотлар билан ишлаганда талабалар йўл қўйиши мумкин бўлган ва энг кенг тарқалган ҳатолик Edit ойнасига киритилаётган маълумот билан боғлиқ. Ёш дастурчилар одатда компьютер билан фақат «ақлли» одамлар ишлайди, улар масала шартida кўрсатилган типдаги маълумотни киритишади деб ўйлашади. (1 ва 2-жадвалларга эътибор беринг) Хўш, бунинг акси бўлса-чи? деган савол одатда уларнинг ҳаёлларига ҳам келмайди. Шунинг учун, одатда улар ёзган дастлабки дастурларнинг таркибida бундай англашилмовчиликлар ва кутилмаган ҳолатлар умуман ҳисобга олинмайди. Ваҳоланки, ёш дастурчилар бундай вазиятларнинг юзага келиб қолиши мумкинлиги ва унинг олдини олишга илк қадамлариданоқ қўнниканлари маъқул.

Талаб қилинган типдаги маълумотни киритиш масаласи ҳам DELPHI тилида ҳимояланган блок механизми try-exception ёрдамида ҳал қилиниши мумкин. Шунинг учун, талабаларни, энг содда, чизиқли дастур ёзишни ўрганганларидан кейиноқ, бу механизм билан таништирилганлиги маъқул. Аммо, бу механизм ҳам дастурда вужудга келиши мумкин бўлган барча ҳатоликларни олишда етарли восита бўла олмайди. Демак, дастурчилар киритилаётган маълумотларни назорат қилишни дастурда ҳисобга олганлари мақсадга мувофиқ бўлади. Намуна тариқасида N! ни ҳисоблаш дастурини кўрамиз. Матнда натурал сонларни киритилишга эътибор беринг.

procedure TForm1.Button1Click(Sender: TObject);

```

var n, i, fak : integer;
begin
  try
    n := strtoint(edit1.text);
  except
    on econvertererror do
      showmessage(' Butun son kirirting');
    end;
    if n<1 then begin
      showmessage('Natural son kirirting');
    exit end
    else
      begin
        fak := 1;

```

```

for i := 1 to n do fak := fak*i;
label1.caption := inttostr(fak);
end;
end;

```

Ушбу матнда *econvertererror* методи *edit1.text* ойнасига киритилган сатрни бутун сонга айлантириш қай даражада амалга оширилганлигини аниқлайди.

Сонли маълумотларни қайта ишлашда энг кўп учрайдиган навбатдаги муаммо - бу олинаётган натижаларнинг шу натижа мансуб бўлган тип диапазонидан четга чиқишидир (1 ва 2-жадвалларга қаранг). Юқоридаги дастур Edit1 ойнасига киритилган 12 сони учун 479 001 600 натижа бергани ҳолда, 13 сони учун 5447020800 ўрнига 1 932 053 504 натижани экранга чиқаради. Бундай катта сонлар билан ишлаш, айниқса олинган натижаларнинг тўғри ёки нотўғрилигини аниқлаш мушкул вазифа ҳисобланади. Бошловчи дастурчилар кўпинча бу кўринишдаги натижаларни тўғри деб қабул қилишади. Чунки, агар улар ёзган дастур нотўғри бўлганда эди, компьютер ҳатолик ҳақида ахборот берган бўлур эди. Компьютер дастурни бажариб, қандайдир натижа бердимиз, демак «дастур тўғри ёзилган» қабилидаги қарашлар ёш дастурчиларни кўпинча хушёрликдан бир оз чалғитади.

Бутун сонлар билан ишлашда бу каби ҳатоликнинг юзага келишининг асосий сабаби – бу маълумотларнинг белгиланган диапазондан четга чиқишидир. Бундай ҳатоликнинг олдини *Erangerror* классига ҳам ола олмайди. Олинган маълумот диапазондан четга чиқса, у ҳолда компьютер ҳатолик ҳақида ахборот бериш ўрнига, олинган натижани тўғридан тўғри [*юқори диапазон+1*] га бўлиб, қолган қолдиқни натижа сифатида қабул қилади.

Дастурда *fak* ўзгарувчисини *longword* типига деб белгилаш ҳам вазиятни қутқара олмайди. Чунки, *n* сони етарлича катта бўлиши мумкин. Яна юқоридаги фикримизга қайтиб, дастурчиларга нафаъат киритилаётган маълумотларни, балки олинаётган натижаларни ҳам дастурда назорат қилишни таъкидлаб ўтамиз. Факториалнинг ҳисоблаш дастурини қуйидагича ёзишга тавсия этамиз.

```

procedure TForm1.Button1Click(Sender: TObject);

```

```

    var n,i, fak: integer;

```

```

begin

```

```

    try

```

```

        n := strtoint(edit1.text);

```

```

    except

```

```

on econverterror do
    showmessage('Butun son kirirting');
end;
if n<1 then begin
    showmessage('Natural son kirirting');
    exit
end
else
begin
    fak := 1;
    for i := 1 to n do begin
        if fak>(2147483647 div i) then
            begin
                showmessage ('Bunday katta sonni yoza olmayman');
                exit
            end
        else begin
            fak := fak*i;
            label1.caption := inttostr(fak);
        end;
    end;
end;
end;
end;
end;

```

Бу матнда 2147483647 сони *longword* типдаги энг катта сон.

Қиймати бутун сонлар диапазонидан четга чиқадиган маълумотлар билан ишлашга эҳтиёж пайдо бўлган, берилган масалани ҳал учун (масалан катта соннинг факториалини ҳисоблаш учун) сатрли маълумотлар билан ишлаш усулларида фойдаланишни тавсия этиш мумкин.

Ҳақиқий типдаги маълумотлар билан ишлаш бутун сонларга қараганда анча содда. Бунинг сабаби шуки, биринчидан, киритилган ихтиёрий сонни ҳақиқий сон сифатида қараш мумкин; иккинчидан, 2-жадвалда кўрсатилган ҳақиқий типдаги ихтиёрий маълумот ўзи мансуб бўлган типнинг юқори диапазонидан четга чикса, бу ҳақда

Floating point overflow

кўринишидаги ахборот чиқарилади. Қуйи диапазондан четга чиқадиган маълумотлар ўрнига эса нол қўйилади.

Бу ҳолда ҳам, зарур бўлса, дастурни ҳақиқий ўрнига ҳақиқий бўлмаган маълумотларни киритишдан ҳимоялаш учун *econverterror* классидан фойдаланиш мумкин.

Экспоненциал кўринишдаги сонларни киритишда эса соннинг мантиссаси учун битта майдон, тартиби учун иккинчи майдон ташкил қилингани маъкул. Шундан кейин бу майдонлардаги маълумотларни

$c := \text{edit1.text} + 'e-' + \text{edit2.Text}; \quad a := \text{strtofloat}(c);$

тарзида экспоненциал кўринишдаги сонга айлантириш мумкин.

Белгили ва сатрли маълумотлар билан ишлаганда одатда, муаммолар кўп бўлмайди. Паскал тилида сатрли катталикларнинг узунлиги 256 тагача бўлиши мумкин эди. DELPHI тилида эса сатрли катталикларнинг узунлиги етарлича катта бўлиши мумкин. Чунки, DELPHI даги *shortstring*, *longstring* ва *widestring* каби сатрли типлар учун хотирадан статик ва динамик тарзда жой ажратилиши мумкин. Бу типлар бир-биридан ўзгарувчилар учун хотирадан жой ажратиш, белгиларнинг кодлаштириш усуллари шамда сатрларнинг узунлиги билан фарқланади. Сатрли ўзгарувчиларга статик жой ажратиш фақат дастур ишга туширилмасдан аввал содир бўлади. Дастур ишга туширилганидан ўзгарувчиларга хотирадан жой динамик тарзда ажратилади. Демак, сатрли маълумотлар учун ажратиладиган жой амалий жиҳатдан компьютернинг бўш хотираси билан чегараланади. Сатрли маълумотлар билан ишлаганда универсал *String* сатрли типи билан ишлашни тавсия этамиз.

Жуда ҳам узун бўлган сатрли катталикларни дастур ёрдамида қайта ишлашга фақат назарий жиҳатдан тўғри келиши мумкин, аммо амалиётда бундай масалалар жуда кам учрайди. Шунинг учун сатрли катталикларнинг узунлиги билан боғлиқ муаммони четга суриб қўйиш мумкин. DELPHI тилида консолли иловалар яратгиш имконияти мавжуд. Аммо, шуни ёдда тутиш керакки, консолли иловаларни Windows муҳитида яратилади ва бу иловалар одатда DOS дастури каби ишлатилади. Биламизки, DOS системасида ASCII кодлаш усули, Windows да эса ANSI усули қўлланади. Бу кодлаш усулларида кирилл алифбесининг ҳарфлари турли кодларга эга. Шунинг сабабли, консолли иловалар дастури матнидаги кирилл алифбесида ёзилган изоҳлар ўрнига бошқа матн экран чиқарилади. Демак, консолли иловаларда турли матнларни чиқаришга зарурат бўлса, бу матнларни лотин алифбесида чиқариш тавсия қилинади.

Агар консолли иловаларда рус алифбесидаги ахборотларни экранга чиқариш зарур бўлса, дастур матнида ANSI-сатрни ASCII-сатрига ўтказувчи қайта кодлаш амалини қўшимча ташкил қилишга тўғри келади. ANSI кодлаш усулида кириллча ҳарфлар 192 дан 255 гача кодланиши ҳамда ASCII да - 128 дан 175 гача (А..Яа..п) ва 224 дан 239 гача (р..я) кодланишини ҳисобга олсак, дастур матнига куйидаги буйруқларни қўшиб қўйиш тавсия қилинади.

```

for i := 1 to length(matn) do
case matn[i] of
'A'..'n' : matn[i] := Chr(Ord(matn[i]) - 64);
'p'..'я' : matn[i] := Chr(Ord(matn[i]) - 16);
end;

```

Биз юқорида фақат энг кўп қўлланадиган бутун, ҳақиқий ва сатрли типларга оид типик маълумотларни келтирдик ҳалос. Қолган типлар билан DELPHI тилида назарий ёки амалий машғулотлар вақтида эҳтиёжга қараб талабаларни таништириш мумкин.

§-5. DELPHI дастурлаш тилида файллар билан ишлаш услугиёти

Маълумки, компьютерларни операцион тизимларсиз, операцион тизимларни файлларсиз тасаввур этиш мумкин эмас. Файл, каталог, диск тушунчалари ихтиёрий тизимнинг ядросини ташкил қилади. Ҳар бир операцион система ўзига ҳос бўлган файллар тизимига эга.

Одатда файллар бажарадиган функцияларига кўра бажариладиган ва бажарилмайдиган турларга бўлинади. Кенгайтмаси **.exe**, **.com**, **.bat** каби файллар бажариладиган файллар ҳисобланади. Бажарилмайдиган файлларнинг кенгайтмаси эса бошқача бўлади. Улар бажариладиган файлларнинг ишини тўлиқ бўлишини таъминлайди ёки улар ёрдамида яратилади. Биз ушбу мавзу ёрдамида бажарилмайдиган файллар гуруҳига кирувчи алоҳида файлларни ўқитиш услугиётини баён қиламиз

Бу файллар маълумотлар омборидан иборат бўлиб, уларни массив тушунчасининг мантиқий давоми деб қараш мумкин. Чунки бу файллар ўзида элементларининг сони олдиндан номаълум ёки чексиз бўлган массивларни сақлайди. Бундай файллар билан ишлашни билиш дастурчилар учун алоҳида аҳамият касб этади ва кейинги фаолиятларида жуда кучли воситага бўлиб хизмат қилади.

Маъруза машғулотларида файл ҳақидаги умумий маълумотлар ҳамда файллар билан ишлашнинг назарий томонларини ўргатилади, амалий машғулотларда эса бу билимларни мустаҳкамлашга интилган ҳолда асосий эътиборни янги файлларни ташкил қилиш, мавжуд файлларнинг давомига янги маълумотларни қўшиш, илгари ташкил қилинган файллар устида турли масалаларни ҳал қилишга қаратилади. Мавзу бўйича мустақил машғулотларни индивидуал топшириқлар асосида ташкил қилиш тавсия қилинади.

Ушбу мавзунинг ёритишдан аввал мавзу бўйича бошланғич маълумот сифатида Файл нима? Файллар билан ишлашнинг афзалликлари нимада?

Файллар билан ишлашга инсоннинг, айниқса компьютер фойдаланувчиларининг ҳаётидаги қандай ҳолатларда эҳтиёж пайдо бўлиши мумкин? деган саволларга жавоб берилиши ҳеч бир шубҳасиз назарий машғулоти самарасини оширади. Бу ўринда намуна тариқасида, қуйидаги ҳолатларни эслатиб қўйиш мумкин.

1-ҳолат: Айрим ҳолларда битта дастурдан фойдаланиш жараёнида катта ҳажмдаги бошланғич маълумотларни киритишга тўғри келиб қолади. Одатда, бу маълумотларни киритиш жараёни бирор сабаб билан тўхтатиб қўйилса ёки узилиб қолса, дастур ўз ишини тўхтатади. Дастур қайта ишга туширилганда эса бошланғич маълумотларни компьютер хотирасига такроран киритиш лозим бўлади. Юзага келиши мумкин бўлган бундай вазият фойдаланувчиларга дастурдан фойдаланишда қийинчиликлар туғдириши мумкин.

2-ҳолат: Фараз қилайлик, компьютер ўз хотирасига киритилган дастурни тўла бажариши учун 1 ой давомида узлуксиз ишлаши керак бўлсин. Ва бу компьютерни қандайдир сабаб билан, айтайлик, 20 кундан кейин ўчиришга тўғри келсин (ёки электр таъминоти маълум бир муддатда узилиб қолсин). Вужудга келиши мумкин бўлган бундай ҳолатда дастурни яна ишга тушириб, компьютернинг дастлабки 20 кунлик меҳнатини қайтадан тиклагандан кейингина дастур ўз ишини давом эттириши мумкин.

Ана шу каби ҳолатларни олдини олиш учун янги иловаларни яратиш жараёнида маълум бир муддат ишлагандан сўнг, дастур ёрдамида олинган натижаларини сақлаб қолиш ва юқоридаги каби кутилмаган узилишлардан кейин дастур ўз ишини келган жойидан давом эттириш имкониятларини назарда тутиш талаб қилинади. Бу ишни ташкил қилиш учун зарур бўлган восита сифатида файлдан фойдаланиш тавсия қилинади.

Файллар маълумот сақлашнинг энг қулай усули эканлигининг сабаби қуйидагилардан иборат:

1) Одатда дастурни бажариб, олинган натижалар дастур ўз ишини тугатгандан сўнг, ЭХМ хотирасидан ўчиб кетади. Бу маълумотларга кейинчалик эҳтиёж пайдо бўлса, уларни қайта тиклаш учун дастурни янгидан ишга туширишга тўғри келади. Буни олдини олиш учун олинган натижаларни файлларга ёзиб қўйилиши мумкин;

2) Битта файлда сақланаётган маълумотлар кўплаб масалалар (иловалар) учун асос бўлиши мумкин, бошқача айтганда сақлаб қўйилган маълумотлар доирасидаги бошқа масалалар учун бошланғич маълумот бўлиб хизмат қилиши мумкин;

3) Маълумотлар сони ЭХМ нинг оператив хотирасига сиғмайдиган даражада кўп бўлиши мумкин. Бундай вақтда маълумотларнинг бир

қисмини қандайдир файлда вақтинча сақлаб қўйиш тавсия қилинади.

Бу маълумотлар тингловчиларнинг қизиқишларини орттириб, файл тушунчаси билан танишишга жисмонан ва руҳан тайёрлайди. Шундан кейин файл ҳақидаги умумий маълумотларни келтиришга ўтиш машғулоти самарасини оширади..

Файл деб, хотира қурилмаларидан бирида сақланаётган (маълум бир ҳажми эгаллаган) ва ўзининг номига эга бўлган маълумотлар тўпламига айтилади.

Файл бўш бўлиши ҳам мумкин.

Ушбу маълумотлар келтирилгандан кейин файлнинг манзили ва номи ҳақидаги умумий тушунчаларни изоҳлашга ўтилади.

Файллар ўзининг манзили ҳамда номига эга бўлади. Файлнинг номи одатда иккита қисмдан иборат бўлади: атамаси ва кенгайтмаси. Масалан:

D: \ DELPHI \ alomat.pas

ёзуви alomat.pas файлини англатади. бу ерда alomat- файлнинг номи, .pas-эса унинг кенгайтмаси. Бу файлнинг манзили бўлиб D дискдаги DELPHI папкаси хизмат қилади.

Машғулотнинг навбатдаги босқичи файлларни ташкил қилиш ҳамда улардан фойдаланишга қаратилади.

Дастлаб, файллар билан ишлаш жараёни икки босқичдан, яъни файлни ташкил қилиш ҳамда файлдан фойдаланиш босқичларидан иборат бўлиши ўқтирилади.

Одатда, файллар билан боғлиқ масалаларнинг шартларида шунчаки бирор файл берилганлиги таъкидланади ва бу файл устида қандайдир амални бажариш талаб қилинади. Масалан: f файлида 1000 гача бўлган туб сонлар сақланади. Шу файлдан фойдаланиб, 500 дан катта бўлган туб сонлар миқдорини аниқланг. Бу каби масалалар тингловчиларни то улар файллар билан ишлаш мавзусини тўлиқ эгаллаб олмагунларича бир оз чалғитади. Оқибатда улар бирданига шундай файл бор экан деган нуқтаи назар билан, тўғридан-тўғри масалада қўйилган талабни бажаришга киришиб кетадилар ва бир оздан сўнг «нотўғри кўча» га кириб қолганларини англаштиради. Бунинг олдини олиш учун дастлаб масаланинг шартида келтирилган файлнинг бирор папкада мавжудлигини текшириш, мавжуд бўлмаганда эса бу файлни яратиш лозимлигини, шундан кейингина бу файл устида талаб қилинган шартни амалга ошириш лозимлигини алоҳида таъкидлаб ўтиш мақсадга мувофиқ.

Юқори келтирилган маълумотларнинг ҳаммаси файллар мавзусини чуқур ўзлаштириш учун бошланғич маълумотлар ҳисобланади. Машғулоти мобайнида бу маълумотлар етарлича изоҳланганидан сўнг, бевосита

файллар устида бажариш мумкин бўлган амаллар ҳақидаги маълумотларга ўтишдан аввал файлли ўзгарувчилар тушунчасини тингловчилар онгига етказиш тавсия этилади.

Одатда файлларда бир хил типдаги маълумотлар сақланади ва улар билан ишлаш учун файлнинг манзили ва номини аниқ кўрсатиш талаб қилинади. Файлларнинг манзили ва номини бир дастур матнида бир неча марта келиши дастурчига бир оз ноқулайлик туғдиради ва ушбу ҳолатнинг олдини олиш учун DELPHI тилида шу файл ва унинг манзилини бирор ўзгарувчи билан белгилаб қўйиш қабул қилинган. Бундай ўзгарувчиларни файлли ўзгарувчилар деб аталади. Файлли ўзгарувчиларни ҳам дастурда қатнашадиган бошқа ўзгарувчилар каби эълон қилиниши лозим. Эълон умумий кўринишда қуйидагича ёзилади:

Var файлли ўзгарувчи: file of mun;

Масалан:

***Var f : file of integer;
g : file of string[20];***

Бу ерда фақат бутун сонлар учун мўлжалланган **f** ҳамда ҳар бир элементи 20 тагача белгидан иборат бўлган матнли маълумотларнинг **g** – файлли ўзгарувчилари эълон қилинмоқда.

Тингловчиларда файлли ўзгарувчилар тўғрисидаги тасаввур пайдо бўлганидан кейин, бу ўзгарувчилар ёрдамида файлларнинг ном ива манзилини кўрсатиш учун мўлжалланган **Assignfile** процедураси ҳақидаги маълумотлар келтирилади. Бу процедура умумий кўринишда

Assignfile(файлли ўзгарувчи, 'файлнинг манзили ва номи');

тарзида ёзилади. Масалан:

assignfile(f, 'D:\ DELPHI \ alomat.pas');

Бундай эълондан кейин, компьютер **f** файли деганда D дискда жойлашган DELPHI папкасидаги alomat.pas файлини тушунади, яъни ҳар гал **f** файли устида бажарилиши лозим бўлган амалларни D:\DELPHI\alomat.pas файли учун бажаради.

Файлларни эълон қилиш тушунтирилганидан сўнг, унинг устида бажариш мумкин бўлган амалларга ўтиш мумкин.

Файллар устида бирор амални бажариш учун аввал уни очиш талаб қилинади. Файлларни очиш эса уч хил мақсаддан фақат биттаси учун амалга оширилади. Файлларни очиш масаласи мақсадга кўра қуйидаги методлар ёрдамида ҳал қилинади:

Rewrite(f). - **f** файлидаги маълумотларни ўқиш;

Reset(f) - **f** файлидаги маълумотларни ўқиш учун очиш;

Append(f) - *f* файлининг охирига янги маълумотлар қўшиш учун очиш.

Дастур матнининг охирида ёки зарурат пайдо бўлганда очилган файлларни албатта ёпиш талаб қилинади. Бунинг учун **Close(f)** - *f* файлини ёпиш методидан фойдаланилади. .

Шундан кейин файллардаги маълумотларни ўқиш ва файлга маълумотларни ёзиш учун мўлжалланган методлар ҳақидаги маълумотларга ўтиш мумкин. Бу ўринда **Write(f, маълумот)**- *f* файлига навбатдаги маълумотни ёзиб қўйиш ҳамда **Read(f, ўзгарувчилар)**- *f* файлидан навбатдаги маълумотни ўқиш ва ўзгарувчиларга қиймат қилиб бериш методларини эслатиб ўтиш кифоя.

Энди очилган файллар бўйлаб курсор тушунчасини киритиш ҳамда унинг ҳаракатини изоҳлаш лозим бўлади. Бунда қуйидаги маълумотларни келтириш фойдадан ҳоли бўлмайди.

Файллар учун курсор-кўрсаткич мавжуд. У доимо қайсидир маълумотни кўрсатиб туради. Файл ўқиш учун очилган бўлса, ундан маълумотларни ўқишда, одатда компьютер курсор кўрсатиб турган маълумотни ўқийди. Ҳар гал белгиланган файлдан навбатдаги маълумот ўқилганидан сўнг, курсор ўз вазиятини ўзгартиради, яъни ўқилган маълумотдан кейинги маълумотга ўтади. Фараз қилайлик, *f* - файлида қуйидаги маълумотлар сақланаётган бўлсин. Агар файл ўқиш учун очилган бўлса, дастлаб кўрсаткич биринчи маълумотни кўрсатиб туради. Ҳамма маълумотлар ўқиб бўлингандан сўнг, кўрсаткич файлнинг охирига ўтади

Файл ёзиш учун очилган бўлса, навбатдаги маълумотлар файлнинг охирига ёзилади.

DELPHI дастурлаш тилига курсорнинг ҳолатини назорат қилувчи **Eof(f)** - *f* файлининг охирги ёзуви ўқилганлигини текшириш ва **Eoln(f)** - *f* файлидаги жорий сатрнинг охирги маълумотини ўқилганлигини текшириш каби мантикий методлар киритилган.

Курсорнинг ҳолати, **Eof(f)** ва **Eoln(f)** каби методларини изоҳлашда анимацион эффектлардан фойдаланиш яхши самара беради.

Шу билан назарий маълумотларни тугатиш мумкин. Навбатдаги вазифа - назарий маълумотларни амалиётга боғлаш масаласидир. Уни конкрет масала ёрдамида ҳал қилиш тавсия қилинади ва албатта файллар билан боғлиқ масалаларни ечиш жараёни одатда иккита босқичдан иборат бўлишини алоҳида эътиборга олиш талаб қилинади:

1. файлни ташкил қилиш;
2. шу файл устида масала шартида берилган амални бажариш.

Кўп йиллик тажрибалар шуни кўрсатадики, тингловчилар масаланинг шартда берилган файлларни компьютерда мавжуд деб қарашади, ваҳолангки бу файл мавжуд бўлмаслиги мумкин. Шунинг учун қўйилган масалани ечишда энг аввало дастлабки шу файлнинг мавжуд ёки мавжуд эмаслигини аниқлаш лозимлигини алоҳида уқтирган маъқул. Агар мавжуд бўлса тўғридан-тўғри масалани ечишга ўтиш мумкин, акс ҳолда дастлаб файлни ташкил қилиш ва шундан кейингина масалани ечишга ўтиш талаб қилинади. Намуна тариқасида келтириладиган масалалар учун иловалар яратишда файллар билан ишлаш жараёнида фойдаланувчилар дуч келиши мумкин бўлган файлни янгидан ташкил қилиш, унинг давомига янги маълумотларни қўшиш ҳамда қўйилган масала шартда талаб қилинган вазифани ҳал қилиш каби муаммоларнинг ҳар бирини алоҳида ҳал қилиш тавсия қилинади.

Назарий машғулотлар самарасини ошириш маърузачидан анчагина куч вақт ва билимни талаб қилади. Бунинг сабаби шундаки, бу мавзу бўйича анъанавий усулда, яъни маъруза шаклида машғулотларни ташкил қилиш яхши натижа бермайди. Чунки, тингловчилар маълумотлар омборини файллар ёрдамида ташкил қилиш билан боғлиқ вазиятларга дуч келишмаган. Шунинг учун ҳам улар файллар ҳақидаги маълумотларни бир қарагандан мураккаб, аммо энг янги, муҳим ва қизиқарли тушунчалардан бири сифатида қабул қиладилар.

Тажрибалар шуни кўрсатдики, файлларни ташкил қилиш, қайта ишлаш, файллардан фойдаланиш, курсорнинг файлдаги ҳолати ҳақидаги дастлабки маълумотлар талабанинг онгига қийинлик билан етиб боради. Бу вазифани осонлаштириш учун маърузачи ўзи танлаган машғулоти ўтказиш услубига мос келадиган файллар билан ишлашни намойиш қилувчи ёки анимацион характердаги ўргатувчи дастурлардан кенг фойдаланиши тавсия қилинади.

МУНДАРИЖА

§-1. Дастурлашнинг янги технологиялари сифатида объектга йўналтирилган дастурлаш технологияларидан фойдаланиш	7
§-2. Объектли алгоритм	15
§-3. Талабаларни DELPHI дастурлаш тилини ўрганишга тайёрлаш босқичлари ҳусусида	22
§-4.Объектга йўналтирилган дастурлаш технологияларини ўқитишда турли типдаги маълумотлар билан ишлаш услубиёти.....	29
§-5 .DELPHI дастурлаш тилида файллар билан ишлаш услубиёти	35

ФОЙДАЛАНИЛГАН АДАБИЁТЛАР РЎЙХАТИ

1. Арипов М. М., Отаханов Н. А. DELPHI дастурлаш тили. 492-бет. Наманган, 2007 й. 492 б.
2. Бабушкина И., С. Окулов. Практикум по объектно-ориентированному программированию. М., Бином, 2004. 366 б.
3. Буч Г. Объектно-ориентированное программирование. М., Конкорд, 2002. 517 с.
4. Н. Вирт. 1989. Алгоритмы и структуры данных. Москва, Мир, 360 стр.
5. С. А. Немнюгин. TURBO PASCAL. Учебник. Москва, Питер, 2002 г. 512 с.
6. Кнут Д. Искусство программирования. М., Мир, 1991.682 с.
7. Тутубалин Д. К.Объекто-ориентированное программирование в среде Delphi: Учеб. пособие.–Томск: Изд-во Том. гос. ун-та систем управл. и радиоэлектрон., 2004. – 112 с.
8. «Программалаш асослари» фанидан намунавий дастур. Тошкент, 2007. 6 б.
9. Stein J. March 1988 . Object-Oriented Programming and Database Design. Dr. Doobs journal of Software Tools for the Professional Programmier. № 137, p.18.
10. Sommerville.1. 1985. Sovtware Engineering. 2nd ed. Wotkhingam England: Addison Wesley , p.68.
11. Yordon E and Konstantine L. 1979. Structured Design. Englewood Cliffs. Nj;Prentice-Hall.

Отаханов Нурилло Абдумаликович

**DELPHI тилини ўқитишдаги
айрим муаммолар ва уларни бартараф
қилиш**

*НамДУ ўқув-услубий кенгашининг
2008 йил 3 июлдаги 10-сонли қарори билан
услубий қўлланма сифатида тавсия этилган*

**Мухаррир: Д. Эргашева
Саҳифаловчи: Е. Савин
Дизайнер: С. Ҳасанов
Мусахҳих: Т. Акбарова**

*Босишга рухсат этилди: 2008 йил 14 июль
Нашириёт босма тобоғи 2,0. Бичими 60x84
Адади 150 дона. Буюртма № 81.
Баҳоси келишилган нархда.*

«ФАРҒОНА» нашриёти
150114, Фарғона шаҳри,
Соҳибқирон Темур кўчаси 28-уй.



Отаханов Нурилло Абдумаликович 1963 йилда Наманган вилояти Чортоқ туманида туғилган. У 1986 йилда Тошкент Давлат Университетининг “Амалий математика ва механика” факультети “Ҳисоблаш математикаси” мутахассислиги бўйича тамомлаб, Наманган Давлат Педагогика институтининг математика факультетига ишга кирган.

Отаханов Н. А. ҳозирги кунда Наманган Давлат университетининг “Амалий математика ва информатика” кафедрасида катта ўқитувчи лавозимида ишлаб келмоқда. У бир нечта илмий ва услубий қўлланмаларнинг муаллифи. Айни вақтда “Объектга йўналтирилган дастурлаш тилларини ўқитиш услубиёти” мавзусида номзодлик ишини олиб бормоқда



«ФАРҒОНА»

Отаханов Нурилло Абдумаликович

**DELPHI тилини ўқитишдаги
айрим муаммолар ва уларни бартараф
қилиш**

*НамДУ ўқув-услубий кенгашининг
2008 йил 3 июлдаги 10-сонли қарори билан
услубий қўлланма сифатида тавсия этилган*

**Мухаррир: Д. Эргашева
Саҳифаловчи: Е. Савин
Дизайнер: С. Ҳасанов
Мусахҳих: Т. Акбарова**

*Босишга рухсат этилди: 2008 йил 14 июль
Наширёт босма тобоғи 2,0. Бичими 60x84
Адади 150 дона. Буюртма № 81.
Баҳоси келишилган нархда.*

«ФАРҒОНА» нашриёти
150114, Фарғона шаҳри,
Соҳибқирон Темур кўчаси 28-уй.



Отаханов Нурилло Абдумаликович 1963 йилда Наманган вилояти Чортоқ туманида туғилган. У 1986 йилда Тошкент давлат университетининг “Амалий математика ва механика” факультети “Ҳисоблаш математикаси” мутахассислиги бўйича тамомлаб,

Наманган давлат педагогика институтининг математика факультетига ишга кирган.

Отаханов Н. А. ҳозирги кунда Наманган давлат университетининг “Амалий математика ва ахборот технологиялари” кафедрасида катта ўқитувчи лавозимида ишлаб келмоқда. У бир нечта илмий ва услубий қўлланмаларнинг муаллифи. Айни вақтда “Объектга йўналтирилган дастурлаш тилларини ўқитиш услубиёти” мавзусида номзодлик ишини олиб бормоқда



«ФАРҒОНА»