

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

ҚАРШИ ДАВЛАТ УНИВЕРСИТЕТИ

«Амалий математика ва информатика» кафедраси

Информатика ва ахборот технологиялари фанидан

РЕФЕРАТ

Мавзу: Алгоритм ва унинг турлари

Бажарди:

**МИС йўналиши 1 курс
талабаси И. Йўлдошев**

Қабул қилди:

Б. Шовалиев

Қарши – 2015 й.

Режа

I. Кириш.

II. Асосий қисм.

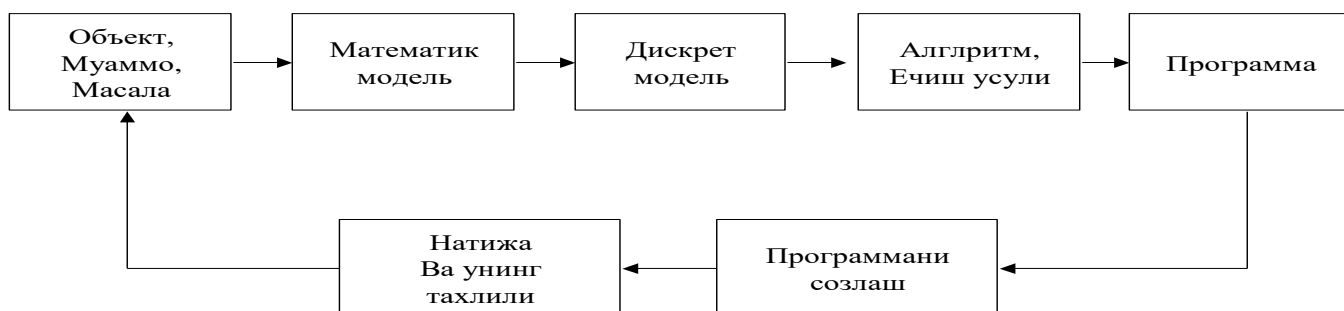
- 1. Алгоритм тушунчаси, унинг асосий хоссалари**
- 2. Алгоритмнинг тасвирлаш усуллари**
- 3. Чизиқли, тармоқланувчи алгоритмлар**
- 4. Такрорланувчи алгоритмлар**

III. Хулоса.

Адабиётлар ва Интернет ресурслар руйхати.

Кириш.

Табиат ёки жамиятда учрайдиган турли муаммо, масала ёки жараёнларни ўрганишни ЭҲМ ёрдамида олиб бориш учун, биринчи навбатда, қаралаётган масала, жараён - объектнинг математик ифодаси, яъни математик моделини, қуриш керак бўлади. Қаралаётган объектнинг математик моделини яратиш жуда мураккаб жараён бўлиб, ўрганилаётган объектга боғлиқ равишда турли соҳа мутахассисларининг иштироқи талаб этилади. Умуман, бирор масалани ЭҲМ ёрдамида ечишни қуйидаги босқичларга ажратиш мумкин.



1-расм. Ҳисоблаш экспериментининг схемаси

Мисол сифатида, космик кемани ердан Зухро планетасига энг оптимал траектория бўйича учуриш масаласини хал қилиш талаб қилинган бўлсин.

Биринчи навбатда, қўйилган масала турли соҳа мутахассислари томонидан атрофлича ўрганилиши ва бу жараённи ифодалайдиган энг муҳим - бўлган асосий параметрларни аниқлаш керак бўлади. Масалан, физик-астроном-инженер томонидан, масала қўйилишининг ўринли эканлиги, яни планеталар орасидаги масофа ва атмосфера қатламларининг таъсири, ер тортиш кучини енгиб ўтиш ва кеманинг оғирлиги, зарур бўлган ёқилғининг оптимал миқдори ва космик кемани қуришда қандай материаллардан фойдаланиш зарурлиги, инсон соғлиғига таъсири ва сарфланадиган вақт ва яна турли туман таъсирларни ҳисобга олган ҳолда шу масаланинг математик моделини тузиш зарур бўлади. Зикр этилган таъсирларни ва физиканинг қонунларини ҳисобга олган ҳолда бу масалани ифодалайдиган бирорта дифференциал ёки бошқа кўринишдаги моделловчи тенглама ҳосил қилиш мумкин бўлади. Балки, бу масалани бир нечта алоҳида масалаларга бўлиб ўрганиш мақсадга мувофиқдир. Бу математик моделни ўрганиш асосида бу масалани ижобий ечиш ёки ҳозирги замон цивилизацияси бу масалани ечишга қодир эмас деган хулосага ҳам келиш мумкин. Бу фикрлар, юқорида келтирилган жадвалнинг 2 блокига мос келади.

Фараз қилайлик биз математик моделни қурдик. Энди уни ЭҲМ да ечиш масаласи туғилади. Бизга Маълумки, ЭҲМ фақат 0 ва 1 дискрет қийматлар ва улар устида арифметик ва мантиқий амалларни бажара олади холос. Шунинг учун математик моделга мос дискрет моделни қуриш зарурияти туғилади (1-расм, 3-блок). Одатда, математик моделларга мос

келувчи дискрет моделлар кўп номаълумли мураккаб чизиқсиз алгебраик тенгламалар системаси (чекли айирмали тенгламалар-схемалар) кўринишида бўлади(4-блок). Энди ҳосил бўлган дискрет моделни сонли ечиш усулини–алгоритмини яратиш зарур бўлади. Алгоритм эса тузиладиган программа учун асос бўлади. Одатда, тузилган программани ишчи ҳолатга келтириш учун программанинг хато ва камчиликларини тузатиш – созлаш зарур бўлади. Олинган сонли натижалар ҳали программанинг тўғри ишлаётганлиги кафолатини бермайди. Шунинг учун олинган натижаларни масаланинг моҳиятидан келиб чиққан ҳолда анализ қилиш керак бўлади. Агар олинган натижа ўрганилаётган жараёни ифодаламаса, масалани 1-расмдаги схема асосида қайтадан кўриб чиқиш ва зарур бўлган жойларда ўзгартиришлар киритиш керак бўлади. Бу жараён, то кутилан ижобий ёки салбий натижа олингунча давом эттирилади ва бу такрорланувчи жараёнга Ҳисоблаш эксперименти деб аталади. Одатда, ҳисоблаш эксперименти деганда соддароқ ҳолда, модел, алгоритм ва программа учлигини (триадасини) тушуниш мумкин.



2 – расм. Ҳисоблаш эксперименти учлиги.

II. Асосий қисм.

2.1. Алгоритм тушунчаси, унинг асосий хоссалари

Юқорида қайд қилганимиздек, қўйилган бирор масалани ЭХМда ечиш учун, аввал унинг математик моделини, кейин алгоритмини ва программасини тузиш керак бўлади. Бу учликда алгоритм блоки муҳим аҳамиятга эга. Энди алгоритм тушунчасининг таърифи ва хоссаларини баён қиламиз.

Алгоритм бу олдимизга қўйилган масалани ечиш зарур бўлган амаллар кетма-кетлигидир.

Масалан квадрат тенгламани ечиш учун қуйидаги амаллар кетма-кетлиги зарур бўлади:

1. a, b, c - коэффициентлар берилган бўлсин,
2. берилган a, b, c - коэффициентлар ёрдамида дискриминант $D = b^2 - 4ac$ ҳисобланади,
3. $D > 0$ бўлса $X_{1/2} = (-b \pm \sqrt{D}) / (2 * a)$
4. $D < 0$ бўлса ҳақиқий ечим йўқ

Мисол сифатида яна берилган a, b, c томонлари бўйича учбурчакнинг юзасини Герон формуласи бўйича ҳисоблаш масаласини кўриб ўтайлик.

1. a, b, c – учбурчакнинг томонлари узунликлари,
2. $p = (a+b+c)/2$ – периметрнинг ярми ҳисоблансин,
3. $T = p(p-a)(p-b)(p-c)$ ҳисоблансин,
4. $S = \sqrt{T}$ ҳисоблансин.

Юқоридаги мисоллардан кўриниб турибдики, алгоритмнинг ҳар бир кадамда бажариладиган амаллар тушинарли ва аниқ тарзда ифодаланган, ҳамда чекли сондаги амаллардан кейин аниқ натижани олиш мумкин.

Зикр этилган, тушинарлилик, аниқлик, чеклилик ва натижавийлик тушунчалари алгоритмнинг асосий хоссаларини ташкил этади. Бу тушунчалар кейинги парарафларда алоҳида кўриб ўтилади.

Алгоритм сўзи ва тушунчаси IX асрда яшаб ижод этган буюк аллома Муҳаммад ал-Хоразмий номи билан узвий боғлиқ. Алгоритм сўзи Ал-Хоразмий номини Европа олимлари томонидан бузиб талаффуз қилинишидан юзага келган. Ал-Хоразмий биринчи бўлиб ўнлик санок системасининг тамойилларини ва ундаги тўртта амалларни бажариш қоидаларини асослаб берган.

Алгоритмнинг 5-та асосий хоссаси мавжуд.

1. **Дискретлилик (Чеклилик).** Бу хоссанинг мазмуни алгоритмларни доимо чекли қадамлардан иборат қилиб бўлаклаш имконияти мавжудлигида. Яъни уни чекли сондаги оддий кўрсатмалар кетма-кетлиги шаклида ифодалаш мумкин. Агар кузатилаётган жараённи чекли қадамлардан иборат қилиб қўллай олмасак, уни алгоритм деб бўлмайди.

2. **Тушунарлилик.** Биз кундалик ҳаётимизда берилган алгоритмлар билан ишлаётган электрон соатлар, машиналар, дастгоҳлар, компьютерлар, турли автоматик ва механик қурилмаларни кузатамиз.

Ижрочига тавсия этилаётган кўрсатмалар, унинг учун тушинарли мазмунда бўлиши шарт, акс ҳолда ижрочи оддийгина амални ҳам бажара олмайди. Ундан ташқари, ижрочи ҳар қандай амални бажара олмаслиги ҳам мумкин.

Ҳар бир ижрочининг бажариши мумкин бўлган кўрсатмалар ёки буйруқлар мажмуаси мавжуд, у ижрочининг кўрсатмалар тизими (системаси) дейилади. Демак, ижрочи учун берилаётган ҳар бир кўрсатма ижрочининг кўрсатмалар тизимига мансуб бўлиши лозим.

Кўрсатмаларни ижрочининг кўрсатмалар тизимига тегишли бўладиган қилиб ифодалай билишимиз муҳим аҳамиятга эга. Масалан, қуйи синфнинг аълочи ўқувчиси "сон квадратга оширилсин" деган кўрсатмани тушинмаслиги натижасида бажара олмайди, лекин "сон ўзини ўзига кўпайтирилсин" шаклидаги кўрсатмани бемалол бажаради, чунки у кўрсатма мазмунидан кўпайириш амалини бажариш кераклигини англайди.

3. **Аниқлик.** Ижрочига берилаётган кўрсатмалар аниқ мазмунда бўлиши зарур. Чунки кўрсатмадаги ноаниқликлар мўлжалдаги мақсадга эришишга олиб келмайди. Одам учун тушинарли бўлган "3-4 марта силкитилсин", "5-10 дақиқа қиздирилсин", "1-2 қошиқ солинсин", "тенгламалардан бири ечилсин" каби ноаниқ кўрсатмалар робот ёки компьютерни қийин аҳволга солиб қўяди.

Бундан ташқари, кўрсатмаларнинг қайси кетма-кетликда бажарилиши ҳам муҳим аҳамиятга эга. Демак, кўрсатмалар аниқ берилиши ва фақат алгоритмда кўрсатилган тартибда бажарилиши шарт экан.

4. **Оммавийлик.** Ҳар бир алгоритм мазмунига кўра бир турдаги масалаларнинг барчаси учун ҳам ўринли бўлиши керак. Яъни масаладаги бошланғич маълумотлар қандай бўлишидан қатъий назар алгоритм шу хилдаги ҳар қандай масалани ечишга яроқли бўлиши керак. Масалан, икки оддий касрнинг умумий маҳражини топиш алгоритми, касрларни турлича ўзгартириб берсангиз ҳам уларнинг умумий маҳражларини аниқлаб бераверади. Ёки учбурчаннынг юзини топиш алгоритми, учбурчакнинг қандай бўлишидан қатъий назар, унинг юзини ҳисоблаб бераверади.

5. **Натижавийлик.** Ҳар бир алгоритм чекли сондаги қадамлардан сўнг албатта натижа бериши шарт. Бажариладиган амаллар кўп бўлса ҳам барибир натижага олиб келиши керак. Чекли қадамдан сўнг қўйилган масала ечимга эга эмаслигини аниқлаш ҳам натижа ҳисобланади. Агар кўрилаётган жараён чексиз давом этиб натижа бермаса, уни алгоритм деб атай олмаймиз.

2.2. Алгоритмнинг тасвирлаш усуллари

Юқорида кўрилган мисолларда одатда биз масалани ечиш алгоритмини сўзлар ва математик формулалар орқали ифодаладик. Лекин алгоритм бошқа кўринишларда ҳам берилиши мумкин. Биз энди алгоритмларнинг энг кўп учрайдиган турлари билан танишамиз.

1. **Алгоритмнинг сўзлар орқали ифодаланиши.** Бу усулда ижрочи учун бериладиган ҳар бир кўрсатма жумлалар, сўзлар орқали буйруқ шаклида берилади.

2. **Алгоритмнинг формулалар билан берилиш** усулидан математика, физика, кимё каби аниқ фанлардаги формулаларни ўрганишда фойдаланилади. Бу усулни баъзан аналитик ифодалаш дейилади.

3. **Алгоритмларнинг график шаклида тасвирланишида** алгоритмлар махсус геометрик фигуралар ёрдамида тасвирланади ва бу график кўриниши блок-схема дейилади.

4. **Алгоритмнинг жадвал кўринишида берилиши.** Алгоритмнинг бу тарзда тасвирланишдан ҳам кўп фойдаланамиз. Масалан, мактабда қўлланиб келинаётган тўрт хонали математик жадваллар ёки турли хил лотереялар жадваллари. Функцияларнинг графикларини чизишда ҳам алгоритмларнинг қийматлари жадвали кўринишларидан фойдаланамиз. Бу каби жадваллардан фойдаланиш алгоритмлари содда бўлган туфайли уларни ўзлаштириб олиш осон.

Юқорида кўрилган алгоритмларнинг тасвирлаш усуллари асосий мақсади, қўйилган масалани ечиш учун зарур бўлган амаллар кетма-кетлигининг энг қулай ҳолатини аниқлаш ва шу билан одам томонидан программа ёзишни янада осонлаштиришдан иборат. Аслида программа ҳам алгоритмнинг бошқа бир кўриниши бўлиб, у инсоннинг компьютер билан мулоқотини қулайроқ амалга ошириш учун мўлжалланган.

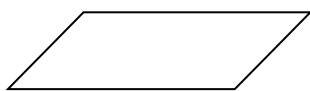
Блок-схемаларни тузишда фойдаланиладиган асосий содда геометрик фигуралар қуйидагилардан иборат.



Овал (эллипс шакли), у алгоритмнинг бошланиши ёки тугаллашини белгилайди.



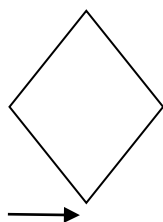
Тўғри бурчакли тўртбурчак, қиймат бериш ёки тегишли кўрсатмаларни бажариш жараёнини белгилайди.



Параллелограмм, маълумотларни киритиш ёки чиқаришни белгилайди.



Ёрдамчи алгоритмга мурожатни белгилайди.



Ромб, шарт текширишни белгилайди ва шарт бажарилса "ҳа", тармоқ бўйича, акс ҳолда "йўқ"-тармоғи бўйича амаллар бажарилишини таъминлайди.

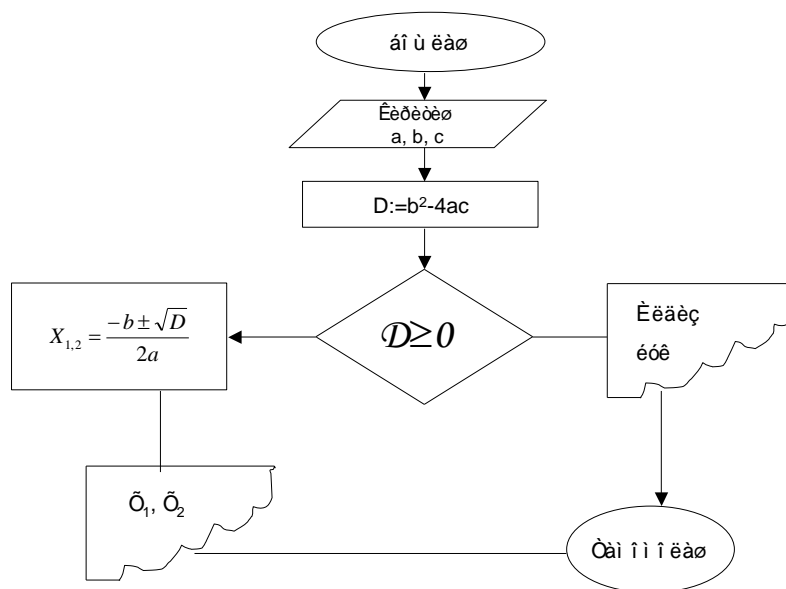
- Стрелка - амаллар кетма кетлигининг бажарилиш йўналишини кўрсатади.

Блок-схемалар билан ишлашни яхшилаб ўзлаштириб олиш зарур, чунки бу усул алгоритмларни ифодалашнинг қулай воситаларидан бири

бўлиб программа тузишни осонлаштиради, программалаш қобилиятини мустаҳкамлайди. Алгоритмик тилларда блок - схеманинг асосий структураларига махсус операторлар мос келади.

Шуни айтиш керакки, блок-схемалардаги ёзувлар одатдаги ёзувлардан катта фарқ қилмайди.

Мисол сифатида 2.1 пунктда келтирилган $ax^2+bx+c=0$ квадрат тенгламани ечиш алгоритмининг блок-схемаси қуйида келтирилган.

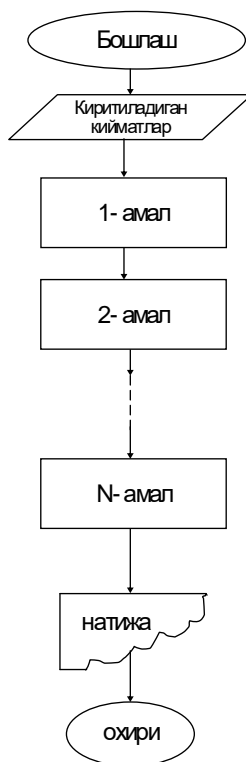


2.3. Чизиқли, тармоқланувчи алгоритмлар

Ҳар қандай мураккаб алгоритмни ҳам учта асосий структура ёрдамида тасвирлаш мумкин. Булар кетма-кетлик, айри ва такрорлаш структураларидир. Бу структуралар асосида чизиқли, тармоқланувчи ва такрорланувчи ҳисоблаш жараёнларининг алгоритмларини тузиш мумкин. Умуман олганда алгоритмларни шартли равишда қуйидаги турларга ажратиш мумкин:

- чизиқли алгоритмлар,
- тармоқланувчи алгоритмлар,
- такрорланувчи ёки циклик алгоритмлар,
- ичма-ич жойлашган циклик алгоритмлар,
- рекуррент алгоритмлар,
- такрорланишлар сони олдиндан ноъмалум алгоритмлар,
- кетма-кет яқинлашувчи алгоритмлар.

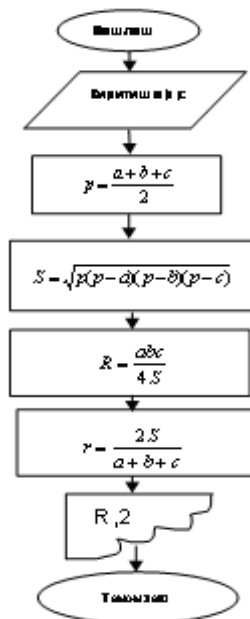
Фақат кетма-кет бажариладиган амаллардан ташкил топган алгоритмларга-чизиқли алгоритмлар дейилади. Бундай алгоритмни ифодалаш учун кетма-кетлик структураси ишлатилади. Структурада бажариладиган амал мос келувчи шакл билан кўрсатилади. Чизиқли алгоритмларнинг блок - схемасини умумий структурасини қуйидаги кўринишда ифодалаш мумкин.



1-мисол. Учбурчак томонларининг узунлиги билан берилган. Учбурчакка ички ва ташқи чизилган айланалар радиуслари ва узунликлари ҳисоблансин.

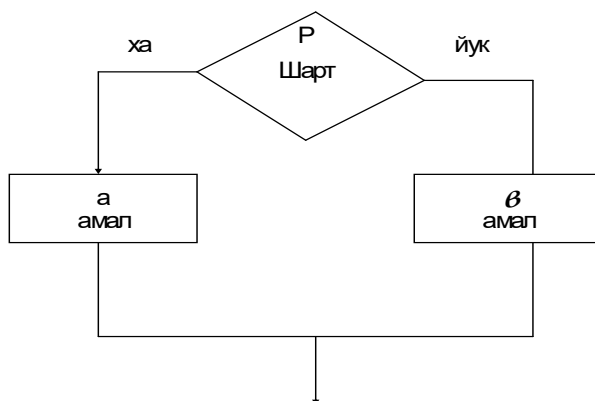
Ички чизилган айлана радиуси $r = \frac{2S}{(a+b+c)}$ ташқи чизилган айлананинг радиуси $R = \frac{4S}{abc}$ формулалар орқали ҳисобланади. Бу ерда S учбурчакнинг юзи, a, b, c -учбурчак томонларининг узунликлари.

Блок-схемани тузамиз.



Тармоқланувчи алгоритмлар.

Агар ҳисоблаш жараёни бирор бир берилган шартнинг бажарилишига қараб турли тармоқлар бўйича давом эттирилса ва ҳисоблаш жараёнида ҳар бир тармоқ фақат бир марта бажарилса, бундай ҳисоблаш жараёнларига тармоқланувчи алгоритмлар дейилади. Тармоқланувчи алгоритмлар учун айри структураси ишлатилади. Тармоқланувчи структураси берилган шартнинг бажарилишига қараб кўрсатилган тармоқдан фақат биттасининг бажарилишини таъминлайди.



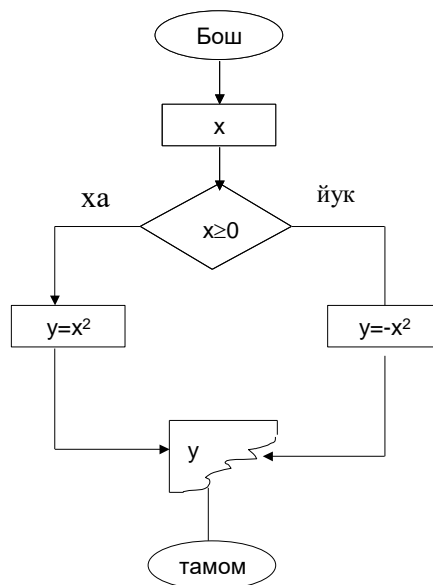
Берилган шарт ромб орқали ифодаланади, р-берилган шарт. Агар шарт бажарилса, "ха" тармоқ бўйича а амал, шарт бажарилмаса "йўқ" тармоқ бўйича в амал бажарилади.

Тармоқланувчи алгоритмга типик мисол сифатида қуйидаги содда мисолни қарайлик.

1- Мисол.

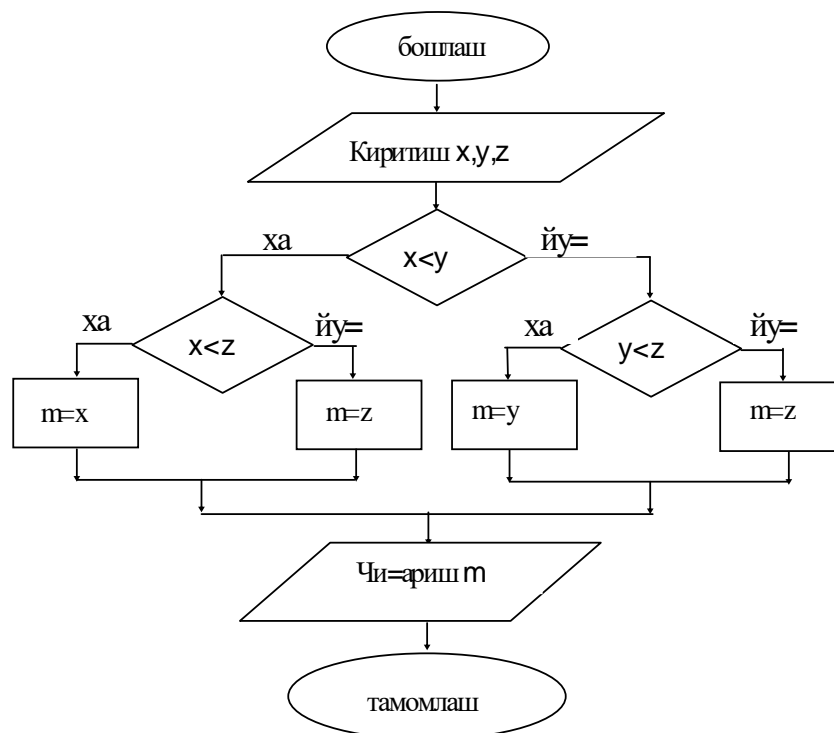
$$Y = \begin{cases} x^2 & \text{агар } x \geq 0 \\ -x^2 & \text{агар } x < 0 \end{cases}$$

Берилган x нинг қийматига боғлиқ ҳолда, агар у мусбат бўлса «ха» тармоқ бўйича $y=x^2$ функциянинг қиймати, акс ҳолда $y=-x^2$ функциянинг қиймати ҳисобланади.

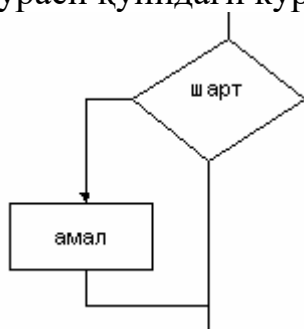


2-мисол. Берилган x, y, z сонлари ичидан энг кичиги аниқлансин. Берилган x, y, z сонлардан энг кичигини m-деб белгилайлик. Агар $x < y$ бўлиб, $x < z$ шарт бажарилса, $m=x$ бўлади, аксинча $x > z$ шарт бажарилса, $m= z$ бўлади. Агар

$x > y$ бўлиб, $y < z$ шарт бажарилса, $m = y$ бўлади, аксинча $y > z$ шарт бажарилса, $m = z$ бўлади. Бу фикрлар қуйидаги блок - схемада ўз аксини топган. Бу блок-схемада тармоқланиш ёки айри структурасидан 3 марта фойдаланилган.



Кўпгина масалаларни ечишда, шарт асосида тармоқланувчи алгоритмларнинг иккита тармоғидан биттасининг яъни ёки «ха» ёки «йўқ» нинг бажарилиши етарли бўлади. Бу ҳолат тармоқланувчи алгоритмнинг хусусий ҳоли сифатида айланиш структураси деб аташ мумкин. Айланиш структураси қуйидаги кўринишга эга:



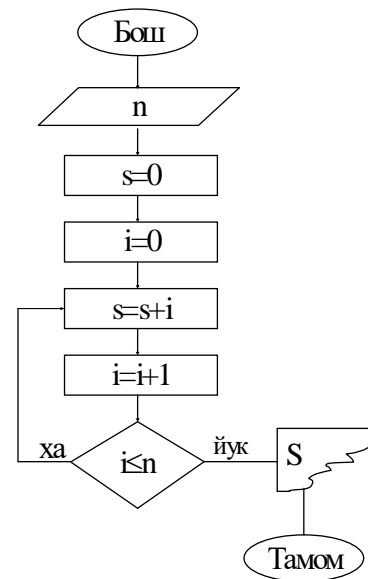
2.4. Такрорланувчи алгоритмлар

Агар бирор масалани ечиш учун тузилган зарур бўлган амаллар кетма-кетлигининг маълум бир қисми бирор параметрга боғлиқ кўп марта қайта бажарилса, бундай алгоритм такрорланувчи алгоритм ёки циклик алгоритмлар дейилади. Такрорланувчи алгоритмларга типик мисол сифатида одатда қаторларнинг йиғиндиси ёки кўпватмасини ҳисоблаш жараёнларини қараш мумкин. Қуйидаги йиғиндини ҳисоблаш алгоритмини тузайлик.

$$S = 1^2 + 2^2 + 3^2 + \dots + N^2 = \sum_{i=1}^N i^2$$

Бу йиғиндини ҳисоблаш учун $i=0$ да $S=0$ деб оламиз ва $i=i+1$ да $S=S+i^2$ ни ҳисоблаймиз. Бу ерда биринчи ва иккинчи қадамлар учун йиғинди ҳисобланди ва кейинги қадамда i параметр яна биттага орттирилади ва навбатдаги рақам аввалги йиғинди S нинг устига қўшилади ва бу жараён шу тартибда то $I < N$ шарти бажарилмагунча давом эттирилади ва натижада изланган йиғиндига эга бўламиз. Бу фикрларни қуйидаги алгоритм сифатида ифодалаш мумкин.

1. N – берилган бўлсин,
2. $i=0$ берилсин,
3. $S=0$ берилсин,
4. $i=i+1$ ҳисоблансин,
5. $S=S+i^2$ ҳисоблансин,
6. $I < N$ текширилсин ва бу шарт бажарилса, 4-сатрга қайтилсин, акс ҳолда кейинги қаторга ўтилсин,
7. S нинг қиймати чоп этилсин.



Юқорида келтирилган алгоритм ва блок схемадан кўриниб турибдики амаллар кетма-кетлигининг маълум қисми параметр i га нисбатан N марта такрорланяпти.

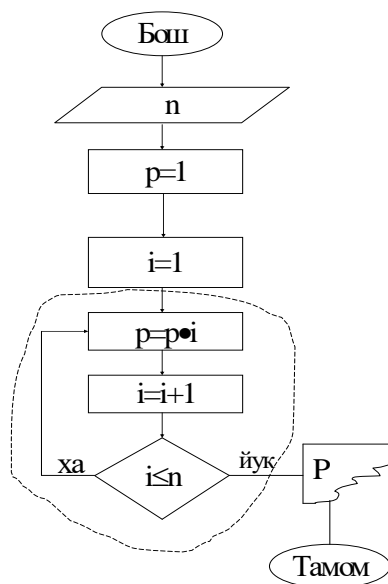
Энди қуйидаги кўпайтманинг алгоритмини ва блок схемасини тузиб кўрайлик. (1 дан N бўлган сонларнинг кўпайтмасини одатда $P!$ каби белгиланади ва факториал деб аталади)

$$P = 1 \cdot 2 \cdot 3 \cdot \dots \cdot N = P!$$

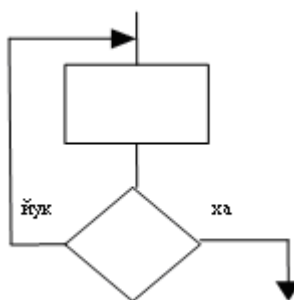
$P!$ - факториални қуйидаги кўринишда ҳам ёзиш мумкин $P = \prod_{i=1}^N i$

Кўпайтмани ҳосил қилиш алгоритми ҳам йиғиндини ҳосил қилиш алгоритмига жуда ўхшаш, фақат кўпайтмани ҳосил қилиш учун $i=1$ да $P=1$ деб оламиз ва кейин $i=i+1$ да $P=P \cdot i$ ни ҳисоблаймиз. Кейинги қадамда i параметр яна биттага орттирилади ва навбатдаги рақам аввалги ҳосил бўлган кўпайтма P га кўпайтирилади ва бу жараён шу тартибда то $I < N$ шарти бажарилмагунча давом эттирилади ва натижада изланган кўпайтмага эга бўламиз. Қуйидаги алгоритмда бу фикрлар ўз аксини топган.

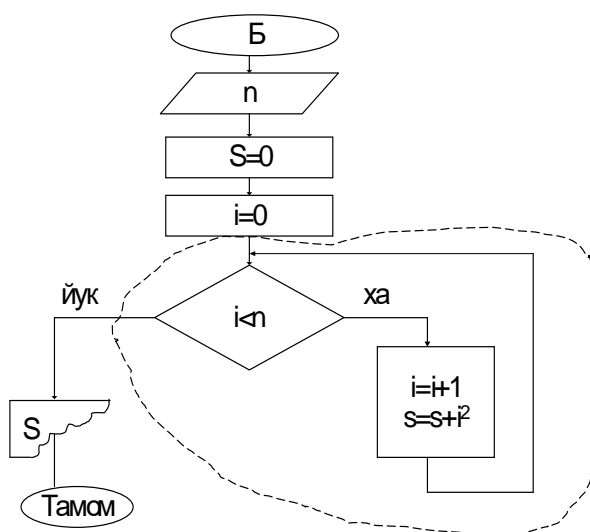
1. N –берилган бўлсин,
2. $i=1$ берилсин,
3. $P=1$ берилсин,
4. $i=i+1$ ҳисоблансин,
5. $P=P * i$ ҳисоблансин,
6. $i < N$ текширилсин ва бу шарт бажарилса, 4-сатрга қайтилсин, акс ҳолда кейинги қаторга ўтилсин,
7. P нинг қиймати чоп этилсин.



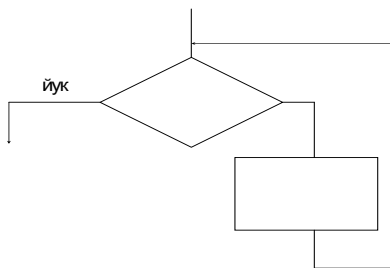
Юқорида кўрилган йиғинди ва кўпайтмаларнинг блок схемаларидаги такрорланувчи қисмларига (айлана олинган) қуйидаги шартни кейин берилган циклик структура мос келишини кўриш мумкин.



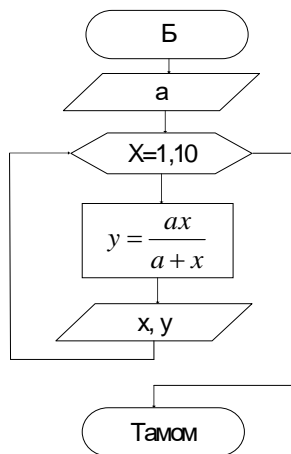
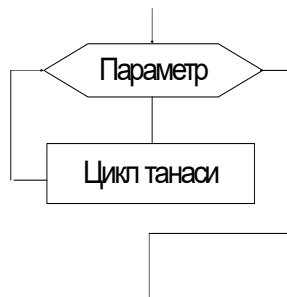
Юқоридаги блок схемаларда шартни олдин текшириладиган ҳолатда чизиш мумкин эди. Масалан, йиғиндининг алгоритмини қарайлик.



Бу блок схеманинг такрорланувчи қисмига қуйидаги, шартни олдин берилган циклик структуранинг мос қилишини кўриш мумкин.



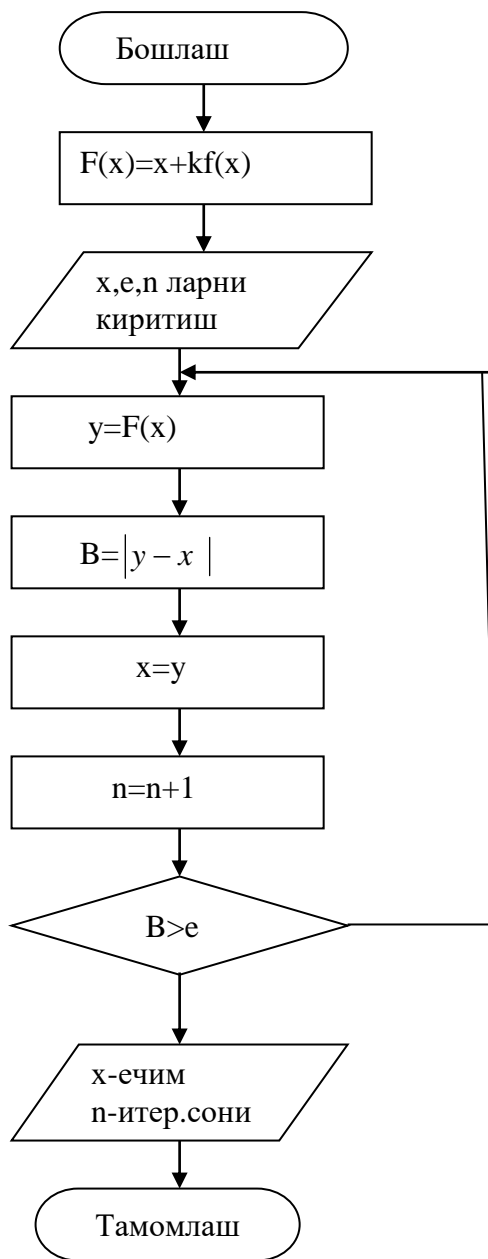
Блок схемаларининг такрорланувчи қисмларини, қуйидаги параметрик циклик структураси кўринишида ҳам ифодалаш мумкин.



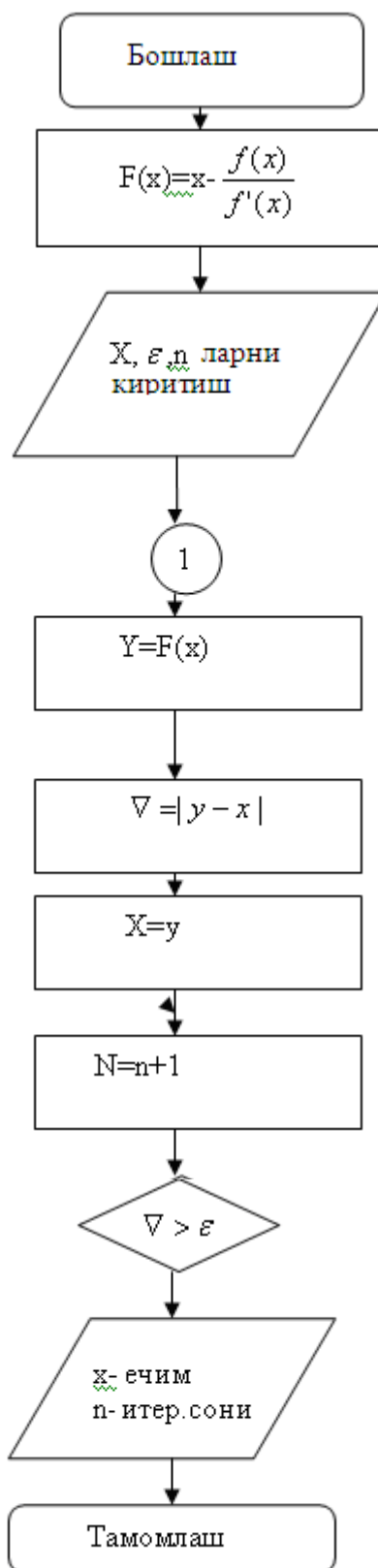
Параметрик цикл структурасига мисол сифатида берилган $x=1,2,3,\dots,10$ ларда $y = \frac{ax}{a+x}$ функциясининг қийматларини ҳисоблаш блок схемасини қараш мумкин.

Илова. Энди баъзи бир масалаларни алгоритмларини куриш жараёнини кўрамиз.

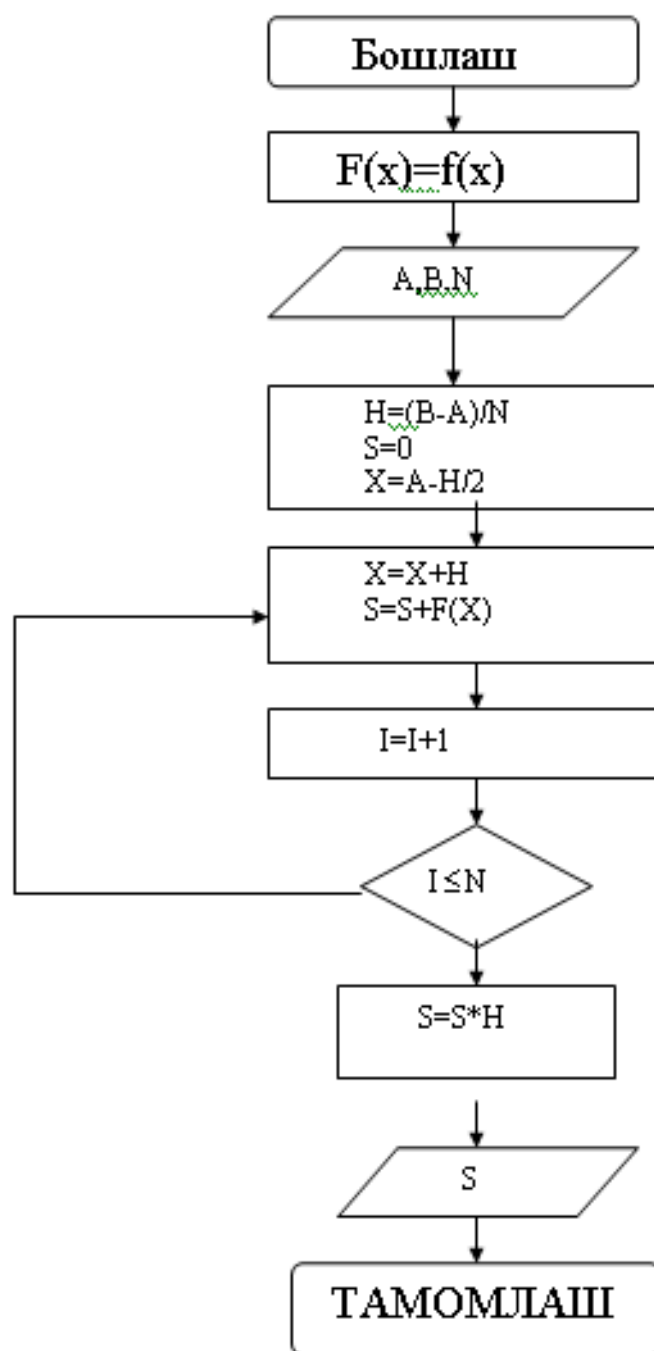
1. Алгебраик тенгламалар системасини оддий итерация усули билан ечиш алгоритми



2. Алгебраик тенгламалар системасини уринмалар (Ньютон) усули билан ечиш алгоритми



3. Аниқ интерални тўғри тўртбурчаклар усули билан тақрибий ҳисоблаш алгоритми



III. Хулоса.

Ушбу рефератни тайёрлашда мавзуга оид материаллар тўпланди ва ўрганиб чиқилди. Ўрганилган материаллар сифатида қуйидагиларни келтириш мумкин:

- ✓ Алгоритм тушунчаси
- ✓ Алгоритмнинг асосий хоссалари
- ✓ Алгоритмнинг тасвирлаш усуллари
- ✓ Чизиқли алгоритмлар
- ✓ Тармоқланувчи алгоритмлар
- ✓ Такрорланувчи алгоритмлар

Рефератнинг охирида баъзи бир масалаларни ечилиш алгоритмларининг блок схемалари келтирилди.

Адабиётлар ва интернет ресурслар руйхати:

1. Халджигитов А.А, Мадрахимов Ш.Ф. Адамбаев У.Э. Информатика ва программалаш. Тошкент 2005 й.
2. Поляков Д.Б., Круглов И.Ю. Программирование в среде Турбо-паскаль. (версия 5.5).М.:МАИ,1992.-576с.
3. Фаронов В.В. Программирование в персональном ЭВМ в среде Турбо-паскаль.-М.:МГТУ,1990. – 443с.
4. Абрамов В.Г., Трифонов Н.П., Трифонова Г.Н. Введение в язык паскаль.-М.:Наука, 1988.-320с.
5. Культин Н.Б. Самоучитель. Программирование в Turbo Pascal 7.0 и Delphi. 2-изд. СПб.:БХВ-Петербург, 2001.-416с.
6. Брябрин В.М. Программное обеспечение персональных ЭВМ. –М.: Наука.,1989.-272с.
7. Бек Л. Введение в системное программирование. –М.:Мир, 1988. – 448с.
8. Донован Дж. Системное программирование.-М.:Мир, 1975.
9. www.intuit.ru
10. www.ziyonet.uz