

Зертханалық жұмыс №1

Есептерді шығаруға C++ тілінің негізгі және қосымша құрылымдарын қолдану. Программалаудың сипатты амалдары

Жұмыс мақсаты: жоғарғы деңгейлі C++ тілінде есептерді шешуге Информатика пәнінде алынған білімдерін практикалық қолдануды үйрену, стандартты алгоритмдерімен практикалық жұмыс жасауға дағдылану.

1.1 Жалпы мәліметтер

1.1.1 Кез келген күрделі алгоритмді үш негізгі басқару конструкцияны қолданып көрсетуге болады.

Негізгі болып келесілер табылады:

- а) *тізбектелу* - әрекеттерді тізбектеп орындауын анықтайды;
- б) *тармақталу* - әрекеттердің екі нұсқасынан біреуін таңдау;
- в) *цикл-әзірше* – орындалуы цикл басында тексерілетін кейбір шарт бұзылғанша қайталанатын әрекеттерді анықтайды.

Негізгілерден басқа программалаудың жоғарғы деңгейдегі процедуралық тілдері негізгі конструкциялар арқылы іске асырылатын үш **қосымша** конструкцияны қолданады:

- а) *таңдау* – кейбір шаманың мәніне қарай бірнеше нұсқадан біреуін таңдау;
- б) *цикл-болғанша* - циклдағы әрекеттері аяқталғаннан кейін орындалуы тексерілетін кейбір шарт орындалғанша қайталанатын әрекеттерді анықтайды;
- в) *қайталану саны берілген цикл* (санаулы цикл) – кейбір әрекеттерді белгіленген сан рет қайталау.

Аталған конструкциялар құрылымдық программалау негізіне салынып, жоғарғы деңгейдегі программалау тілдерінде оларды іске асыру үшін басқару операторлары пайда болды.

Құрылымдық программалардың алгоритмдерін бейнелеу үшін келесілер қолданылады:

- 1) *блок-сұлбалар*;
- 2) *псевдокодтар*;
- 3) *Flow-формалар*;
- 4) *Насси-Шнейдерман диаграммалары*.

А.1 кесетеді алгоритмдерді бейнелеудің әртүрлі амалдарының сәйкестігі көрсетілген.

1.1.2 Басқаруды тапсырудың тек қана құрылымдық операторларын қолдану жолымен жазылған программаларды **құрылымдық** деп атайды. Олардың басқаруды тапсырудың төменгі деңгейлі амалдарын қолданып өңделетін программалардан айырмашылығын көрсету үшін осы атау қолданылады.

Қарастырылған әр басқару конструкциясы үшін C++ тілінің операторын сәйкестіруге болады. Мысалы, негізгі деп келесілерді атауға болады:

- а) *меншіктеу операторы* **айнымалы = өрнек**;

б) *шартты көшу* операторы:

- толық түрі **if (шарт) оператор1; else оператор2;**

- қысқаша түрі **if (шарт) оператор;**

в) *алдын-ала қойылған шарты бар цикл операторы*

```
while (шарт)           немесе       while (шарт)  
{  
    оператор1;  
    оператор2;  
    операторN;  
}
```

Қосымша конструкцияларға келесілер сәйкестіріледі:

а) *ауыстырып қосқыш* **switch** операторы (*таңдау* немесе *нұсқа* операторы)

```
switch (өрнек)  
{  
    case n1 : 1оператор; break;  
    case n2 : 2оператор; break;  
    case nk : 0Коператор; break;  
    default : Nоператор; break;  
}
```

б) *үш шарт* операторы

(1өрнек) ? 2өрнек: 3өрнек;

в) *соңынан тексерілетін шарты бар цикл операторы*

```
do                                немесе do {  
    оператор;  
while (шарт);                     1оператор;  
                                2оператор;  
...  
                                Nоператор;  
                                } while (шарт);
```

г) *параметрі бар цикл операторы*

for (инициализациялау; тексеру; өсімше)

{1оператор;

2оператор;

...

Nоператор; }

1.1.3 Өртүрлі есептеулерді орындағанда жиі жағдайларда бір-ақ стандартты алгоритмдер қолданылады, мысалы, кейбір құрылымның қосындысын, көбейтіндісін, элементтер санын, максималды немесе минималды мәнін табу есептерінде. Осы алгоритмдер программалаудың сипатты амалдары деп есептеледі. Практикада оларды іске асыру келесі жолмен орындалады:

- циклды ашу алдында ізделінетін параметрдің болжанатын немесе бастапқы мәні орнатылады;

- цикл ішінде жинақталу немесе ізделіну орындалады.

Программалудың сипатты амалдарын практикалық қолданудың кепілдемелері А.2 кестеде келтірілген.

1.2 Зертханалық жұмысқа тапсырма

1.1 кесте бойынша нұсқаны таңдап, негізгі және қосымша конструкцияларды қолдану жолымен есепті шығарыңыз. Пайдаланушымен диалогты ұйымдастырыңыз.

1.1 кесте – Тапсырма нұсқалары

Нұсқа	Тапсырма
Есептеңіз:	
1	Клавиатурадан еңгізілген бөлшек сандар тізбегінің орта арифметикалық мәнін; сандар көлемін пайдаланушы орнатады.
2	Нүктенің қозғалу теңдеуі келесі түрде берілген: $\ddot{x} + \gamma \dot{x} + \omega_0^2 x = F \cos(\omega t)$ Нүкте максималды жылдамдылығына жететін уақыттың t моментін.
3	$(0, 100)$ диапазонында генерацияланған k кездейсоқ сандарының тізбегіндегі минималды мәнін.
4	Мүшелер саны көп болған кезде қосындысы $\pi/4$ мәніне түйісетін $\sin x + \sin 2x + \sin 3x + \dots + \sin nx$ қатарын қолданып, пайдаланушы берген дәлдікпен π санын.
5	$(0, 10)$ диапазонында генерацияланған 10 кездейсоқ сандарының тізбегіндегі орта геометриялық мәнін, сандарды экранға шығарыңыз.
6	Клавиатурадан еңгізілген санның факториалын.
7	$S_n = \sum_{k=1}^n \frac{1}{k^2}$, n – клавиатурадан еңгізілген бүтін сан.
8	Жалға алушының бастапқы капиталы – k теңге, ай сайынғы табысы – $n\%$, жәйді жалға алу – m теңге. Өз жәйін сатып алуға жалға алушы неше жылдан кейін S сомасын жинайды.
9	$a_0 = -20$, $a_i = a_{i-1} + i/2$ болатыны белгілі. Осы тәртіп бойынша жасалған тізбектің бірінші оң таңбалы элементін.
10	Екі бүтін сандарының ең үлкен жалпы бөлгішін.

1.1 кестенің жалғасы

11	(0, 50) диапазонында генерацияланған k кездейсоқ сандарының тізбегіндегі максималды мәнін.
12	Нүктенің қозғалу теңдеуі келесі түрде берілген: . Нүкте максималды үдеуіне жететін уақыттың t моментін.
13	$a_0 = 15$, $a_i = a_{i-1} - i/2$ болатыны белгілі. Осы тәртіп бойынша жасалған тізбектің бірінші теріс таңбалы элементін.
14	(-100, 100) диапазонында генерацияланған N кездейсоқ сандарының тізбегіндегі бірінші саннан үлкен болатын сандар көлемін. Барлық сандарды экранға шығарыңыз.
Экранға шығарыңыз:	
15	2-нің дәрежелер кестесін. Дәреженің n мәні пайдаланушымен анықталады.
16	Кез келген батырманы басқанша жұмыс істеп тұрған «электронды сағатты».
17	Пайдаланушы анықтайтын n санының көбейту кестесін.
18	Бірінші m бүтін оң таңбалы сандарының квадраттарының кестесін.
19	Берілген уақыттың t аралығы біткеннен кейін дыбыс сигналды беретін таймерді. t мәні клавиатурадан енгізіледі.
20	Шахмат тақтаның бейнесін: қара торларды жұлдызшамен, ақ торларды – бос орынмен белгілеңіз.
21	Z санның дәрежелер кестелерін (0 дәрежеден k -ға дейін); k мәнін пайдаланушы орнатады.
Тексеріңіз:	
22	x аргументі $[0; 5T]$ интервалында өзгеріп, периоды $T=2\pi$ болғанда функциясы периодты бола ма?
23	Пайдаланушының көбейту кестесін білуін: k есептерді шығарғызып, баға қойыңыз (дұрыс есептер 90-100% болған кезде бағасы - «өте жақсы», 75-89% - «жақсы», 55-74% - «қанағат», 55% төмен - «қанағат емес»).
24	(1,10) диапазонында компьютер генерациялаған бүтін санды 5 талапта болжап білуді пайдаланушыға ұсыну.
25	Пайдаланушы енгізген k бүтін сандар жәй сандар болады ма?
Теңдіктердің дұрыс екенін дәлелдеңіз. Ол үшін пайдаланушымен берілген x аргумент мәнінде сол жақтағы функцияны және оң жақтағы тізбекті берілген дәлдікпен есептеніз:	
26	

1.1 кестенің соңы

27	
28	
29	
30	

1.3 Бақылау сұрақтары

1.3.1 Алгоритмдік құрылымдардың қандайы негізгі болып табылады? Қандайы қосымша болады? Негізгі және қосымша құрылымдардың мысалдарын келтіріңіз.

1.3.2 Алгоритмдердің сұлбалары графикалық түрде қалай бейнеленеді? Flow-формалар мен Насси-Шнейдерман диаграммаларының артықшылықтары не де? Блок-сұлбалардың артықшылықтары мен кемшіліктері?

1.3.3 Оператор мен операция арасындағы айырмашылық? Операциялардың қандай түрлері бар? C++ тіліндегі меншіктеу операцияларды атаңыз.

1.3.4 Форматты енгізу – шығарудың ерекшеліктерін атаңыз.

1.3.5 C++ тілінің қандай операторлары тармақталуды іске асырады? Ерекшеліктері не де? Пайдаланудың мысалдарын келтіріңіз.

1.3.6 Таңдау оператордың құрылымы қандай? Қандай мақсатымен *break* операторы қолданылады?

1.3.7 Циклдарды C++ тілінің қандай операторлары іске асырады? Ерекшеліктері не де? Пайдаланудың мысалдарын келтіріңіз.

1.3.8 *while* циклы көмегімен қалай **for** циклын имитациялауға болады?

1.3.9 $a_1, a_2, a_3 \dots a_{10}$ тізбегінің элементтерінің орта арифметикалық мәнін табу алгоритмін келтіріңіз. Осы тізбектің орта геометриялық мәнін іздеу алгоритмінің қандай айырмашылығы болады?

1.3.10 $a_1, a_2, a_3 \dots a_{10}$ тізбегінің элементтерінің максималды мәнін табу алгоритмін келтіріңіз.