

КІРІСПЕ

Осы нұсқаулықтарды әзірлеу кезінде алдыңғы жылдары оқыту тәжірибесі «Алгоритмдеу және программалау тілдері» және «Алгоритмдік тілдерде бағдарламалау» қолданылған.

Нұсқаулық теориялық материалды, көп көлемде есептерді шешу мысалдарын, жеке тапсырмалар бойынша СӨЖ нұсқалары және сұрақтарды, зертханалық жұмысты қорғау үшін СОӨЖ тапсырмаларын қамтиды.

Студенттердің өздік жұмысына арналған нұсқаулықта әрбір пән үшін дәріс материалын алдын ала зерттеу ұсынылады. Зертханалық жұмысқа арналған нұсқаулықты оқу компьютермен орындау кезінде практикалық игеру үшін қажет. Алдын ала дайындықсыз тақырыптың тест сұрақтарына және үй тапсырмасын орындау қиынға соғады .

Осы әдістемелік нұсқаулықпен жұмыс жасай отырып, Сіз оқып жатқан тақырыпқа арналған тест сұрақтарына жауап беру керексіз. Егер кейбір сұрақтар қиындық туғызса, онда Сіз теориялық материалды тақырыптың тиісті бөлімінен қайталауыңыз керек. Егер сізге теориялық материалды түсіну қиын болса, онда сіз СОӨЖ сағатында мұғалімнің қосымша сабағына келуіңіз керек.

Тест сұрақтарына жауап бергеннен кейін оқып жатқан тақырыптың үй тапсырмасын орындауға кірісіңіз. Бірінші қадамда Сіз үй тапсырмасының талаптарын түсініп және оның орындалу алгоритмін әзірлеуіңіз керек. Екінші қадамда Сіз бағдарламаны әзірлеуіңіз керек, компьютерде іске қосып және оған тестілеу жүргізу қажет. Аяқталған үй тапсырмасын мұғалімге көрсету үшін дайындау керек.

Үй тапсырмасын табысты аяқтағаннан кейін оқып жатқан пән модулінің жеке тапсырмасын орындауға тырысыңыз.

Тізімделген әрекеттер ретін әрбір оқылатын пән тақырыбы үшін өз бетімен орындау керек!

Осы әдістемелік нұсқаулық лекция материалдарын толықтырады, бірақ пәннің барлық аспектілерінің сипаттамасы болып табылмайды. Сондықтан студенттерге әдебиеттер тізімі ұсынылған, олар оны пәннің материалын тереңірек игеру үшін пайдалана алады.

ТАҚЫРЫП 1 C# БАҒДАРЛАМАЛАУ ТІЛІНІҢ ҚАРАПАЙЫМ ОПЕРАТОРЛАРЫ

1.1 Бірінші тақырыптың мақсаты

Консоль қосымшаларын құру үшін Visual Studio.NET ортасының негіздерімен танысу және практикалық дағды алу. C# тілінің қарапайым операторларын қолданып бағдарламалау тілін игеру.

1.2 Теориялық мәліметтер

1.2.1 Кіріспе

C # тілі программалау тілдері саласында ең танымал тіл болып табылады. Бұл 21-ші ғасырда құрылған бірінші бағдарламалау тілі болып табылады. Тіл халықаралық қоғамдастық тарапынан танылаған. 2006 жылдың маусым айында , Стандарттау жөніндегі еуропалық қауымдастық тіл стандартының төртінші нұсқасын бекітті: Standard ECMA-334 C# Language Specifications, 4-th edition.

C# тілін жасаушылар тобын басқарушы Microsoft қызметкері Андреас Хейлсберг. Ол программистер әлемінде Microsoft келгенге дейін атақты болды. Хейлсберг Delphi визуалды бағдарламалау тілінің жетекші жасақтаушы болды.

C# тілінің келесі негізгі артылықшылықтар белгілеуге болады:

- C # тілі Framework.Net жақтауымен қатар құрылып, дамытылған және оның барлық мүмкіндіктерін толық ескереді;
- C # тілі толық объектілі - бағдарланған тіл болып табылады;
- Framework.Net жақтауының арқасында бағдарламашылар C# тілін виртуалды машинада жұмыс істеу артықшылығын алады;
- Framework.Net C # тілінің қолданбаларының түрлерін қолдайды ;
- іске асыру , сенімді және тиімді код құрылысын үйлестіре отырып, C # тілі табысты ықпал ететін маңызды фактор болып табылады .

1.2.2 C# тілінің типтер жүйесі

C# тілінің стандартына келесі негізгі типтердің жиыны кіреді:

- логикалық типтер (bool);
 - символдық типтер (char);
 - бүтін типтер. Бүтін типтер келесі үш өлшемдердің біреуі бола алады - short, int, long, signed немесе unsigned дискрипторымен бірге, онда мәнді түсіндіре көрсетілген - белгісі жоқ немесе бар ;
 - нақты типтер. Бұл типтер келесі үш өлшемдердің біреуі бола алады - float, double, long double;
 - void типі, ақпараттың жоқ болуды белгілеу үшін қолданылады.
- Тіл түрлерді жобалауға мүмкіндік береді:
- көрсеткіштер (мысалы, int* - айнымалы типіне көрсеткіш int);
 - сілтемелер (мысалы, double& - айнамамы типіне сілтеме double);

- массивтер (мысалы, `char[]` – элемент типінің массиві `char`).
- Тіл пайдаланушы анықтаған түрлерді жобалауға мүмкіндік береді :
- анықталған түрлер (`enum`) нақты жиынтық мәндерін ұсынуға арналған;
- құрылым (`struct`);
- кластар (`class`).

C# тілінің негізгі типтерінің сипаттамасы 1.1 кестесінде көрсетілген.

1.1 Кесте C# тілінің деректерінің негізгі типтері

Типтің аты	Жүйелік тип	Мәні	Өлшемі
<code>bool</code>	<code>System.Boolean</code>	<code>true, false</code>	8 бит
<code>sbyte</code>	<code>System.SByte</code>	<code>[-128, 127]</code>	Белгімен, 8-бит
<code>byte</code>	<code>System.Byte</code>	<code>[0, 255]</code>	Белгісіз, 8-бит
<code>short</code>	<code>System.Short</code>	<code>[-32768, 32767]</code>	Белгімен, 16-бит
<code>ushort</code>	<code>System.UShort</code>	<code>[0, 65535]</code>	Белгісіз, 16-бит
<code>int</code>	<code>System.Int32</code>	$[-2^{31}, 2^{31}]$	Белгімен, 32-бит
<code>uint</code>	<code>System.UInt32</code>	$[0, 2^{32}]$	Белгісіз, 32-бит
<code>long</code>	<code>System.Int64</code>	$[-2^{63}, 2^{63}]$	Белгімен, 64-бит
<code>ulong</code>	<code>System.UInt64</code>	$\approx (0, 2^{64})$	Белгісіз, 64-бит
<code>float</code>	<code>System.Single</code>	$[10^{-45}, 10^{38}]$	7 сан
<code>double</code>	<code>System.Double</code>	$[10^{-324}, 10^{308}]$	15-16 сан
<code>decimal</code>	<code>System.Decimal</code>	$[10^{-28}, 10^{28}]$	28-29 айтарлықтай сандар
<code>char</code>	<code>System.Char</code>	<code>[U+0000, U+ffff]</code>	16-бит Unicode символ
<code>string</code>	<code>System.String</code>	Unicode символдардан тұратын жол	

1.2.3 Өрнектер

Өрнектер операндтар, тұрақтылар, айнымалылар, біріккен операция белгілері бар функциялар мен жақшалардан құралады. Өрнекті есептеу кезінде оның мәні мен түрі анықталады. Бұл өрнек сипаттамалары операндтар мәні және түрімен анықталады және өрнек ішіне кіретін есептеу қағидаларымен де. Ереже қойылады:

- операциялар басымдылығымен (операциялар үшін бір басымдылығын қолдану тәртібі – солдан оңға қарай немесе оңнан солға қарай);
- операндтар түрінің түрленуі және шамадан тыс жүктелген операцияларды іске асыру;
- орындалу операциясының нәтижесінің түрі мен мәні операндқа берілген мәнмен анықталған.

Операцияның басымдылығы мен орындау тәртібі кестеде келтірілген 1.2.

Кесте 1.2 C#-та операцияның басымдылығы мен орындау тәртібі

Басымдық	Категория	Операция	Реті
0	Негізгі	x.y, f(x), a[x], x++, x--, new	→
1	Унарлы	+, -, !, ++x, --x, (T)x	→
2	Мультипликативтік	*, /, %	→
3	Қосымша	+, -	→
4	Ауысым	<<, >>	→
5	Қарым-қатынас түрі тексеру	<, >, <=, >=, is, as	→
6	Эквиваленттік	==, !=	→
7	Логикалық ЖӘНЕ (AND)	&	→
8	Логикалық НЕМЕСЕ (OR)		→
9	Шартты логикалық ЖӘНЕ	&&	→
10	Логикалық шартты немесе		→
11	Шартты өрнек	? :	←
12	Меншіктеу	=, *=, /=, %=, +=, -=	←

Кез келген жақшаға алынған өрнек жоғары басымдылық алады және есептелген болуы тиіс. Қажетті нақты жағдайда жақшалар өрнекті есептеудің стандартты тәртібін өзгертуге және орнатуға мүмкіндік береді. Күрделі өрнектерде жақша қою пайдалы, себебі стандартты тәртіп талап етілген тәртіппен сәйкес келді.

1.2.4 Типтердің түрленуі

Операция кастинга – операндтарды бір типке келтіру. Өрнекті есептеу барысында операнд типтерінің түрленуін орындау қажеттігі туындауы мүмкін. Мүмкіндігінше бұл түрлендіру автоматты түрде орындалады, бағдарламалаушы үшін емес. Бірақ айқын емес түрлендіру шектелген, өйткені қауіпсіз ғана болуы мүмкін. Қауіпті түрлендіруді қашан орындай екендігін бағдарламалаушы анық білуі тиіс. Тапсырма типінің түрленуінің мүмкіндігі операцияны қолдану, ол да кастинг деп аталады. Бұл унарлы операция келесі синтаксиспен:

(тип) өрнек;

Мұнда жақша ішінде тип көрсетілген, жақша сыртында тұрған өрнекті қою керек. Операция кастингін тек арифметикалық түрдің ішінде ғана қолдану керек. Оның көмегімен бір арифметикалық типті басқа бір типке келтіруге болады, бірақ бүтін типті логикалық bool типіне келтіруге болмайды.

Типке келтіру мысалын қарастырайық:

```
byteb1 = 1, b2 = 2, b3;
```

```
//b3 = b1 + b2;
```

```
b3 = (byte)(b1 + b2);
```

Бұл мысалда byte типті екі айнымалыны қосу керек, бір қарағаннан, ешқандай типке келтіру керек жоқ сияқты, қорытындыда да тип byte, тағайындау оператордың сол жағында келісілген. Алайда, бұл қысқа нөмірлерге ешқандай қосу операция бар екені қарапайым себеппен

көрсетілген. Қосуды іске асыру `int` типінен басталады. Сондықтан қосуды орындау алдында екі операндта `int` типіне алмастырылады, қосу қорындысыда типі `int` болады, өрнекке мәнді ауыстыру кезінде `byte` типті айнымалыға ауысу кезінде компиляция периондында қате туындайды. Сол себепті екінші жолдағы оператор коды оқылмайды. Программист өрнекті `byte` типіне келтіруі керек , ол кодтың үшінші жолында келтірілген.

Келесі үзінді кодында типке келтіру мысалы көрсетілген:

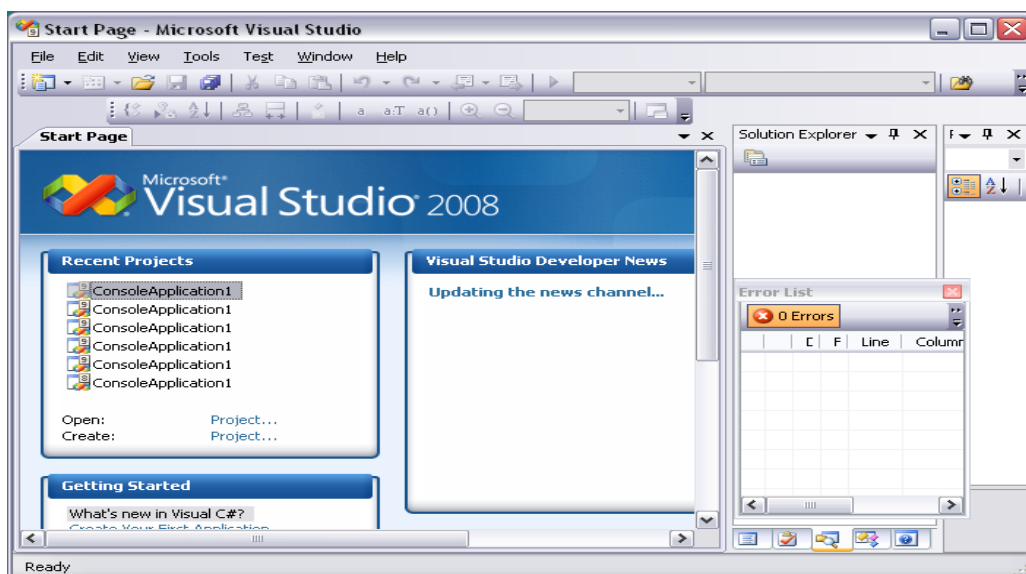
```
int i1,i2;
i1 = -40;
i2 = (int)(1.8 * i1) + 32;
```

Бірінші операнд типінде көбейту қорытындысының типі `double`. Көбейтуді орындамас бұрын, қорытынды `int` типіне келтіріледі. Бірінші көбейтуге келтірген соң `(int)(1.8 * i1)` көбейту нақты сандармен орындалады қорытынды `int` типті болады, `i2` айнымалысына келтіру үшін ешқандай өзгерту қажет емес. Егер операндтағы келтіруді алып тастаса, онда компиляция этапында қателік туындайды.

1.3 Зертханалық жұмысты орындау мысалы

а және b айнымалыларының орын ауыстыратын бағдарлама жазу. А айнымалысының бастапқы мәні 0-ден 20-ға дейінгі диапазонда кездейсоқ түрде құрылады, ал b айнымалысының мәні диалог режімінде енгізіледі. Екі айнымалының орны ауысқаннан кейін шешімін қарау. Бағдарламаны консоль қосымшасында орындау керек және жұмыс үстеліне орыналастыру қажет.

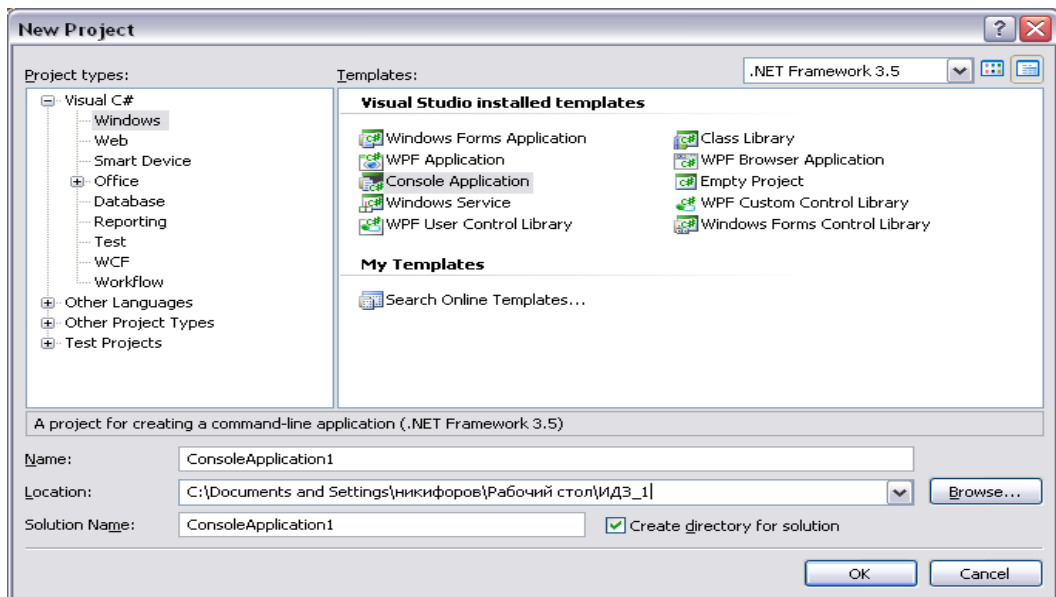
Visual Studio.NET бағдарламалау ортасын іске қосамыз (1.1 суретке қара).



1.1 Сурет – 1 терезе – Visual Studio.NET жүктеу

1.1 суреттің терезесінде File/New/Projekt... командаларды таңдаймыз

Жаңа терезе ашылады (New Project), онда жасақталатын жобаның типін таңдауға болады (1.2 суретті қара).



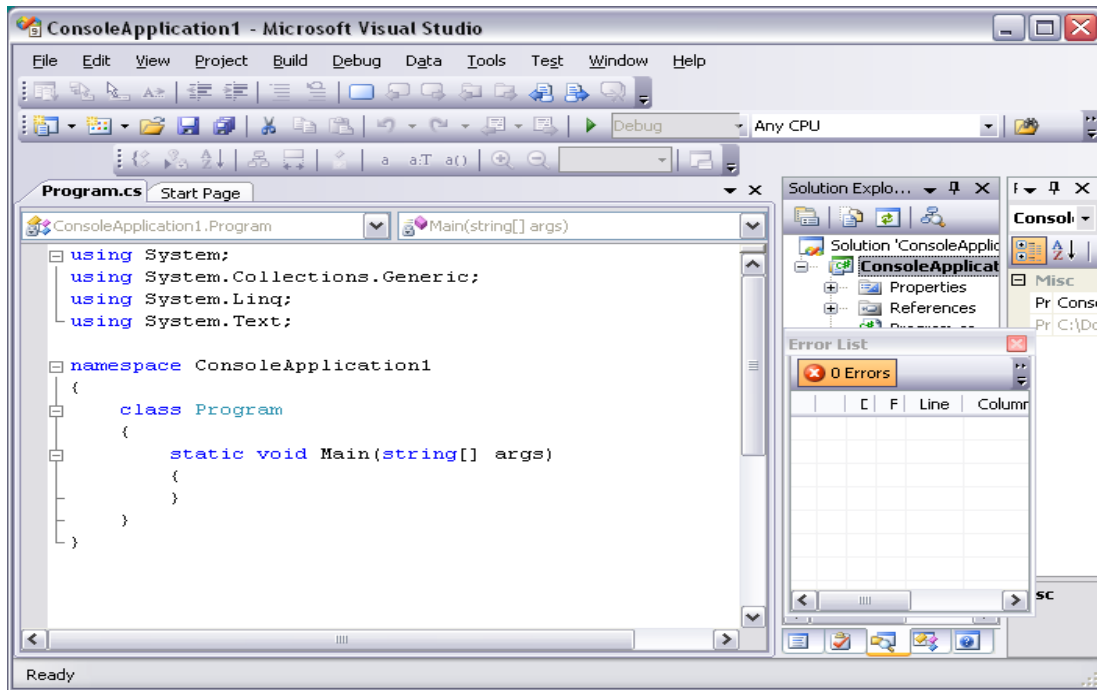
1.2 сурет – 2 Терезе – жаңа жобаны құру

2 терезеде

Калған терезе настройкаларын өзгермеуге болады.

2 терезеде жұмыстың типін ConsoleApplication таңдаймыз және и сақтау орынын көрсетеміз, ол жерде жұмыстың құрылған файлдары сақталады (Рабочий стол папка ИДЗ_1). Терезенің басқа баптауларын өзгертпеуге болады.

ОК батырманы басамыз және келесі 3 терезеге көшеміз.



1.3 сурет – 3 Терезе Visual Studio.NET ортасы

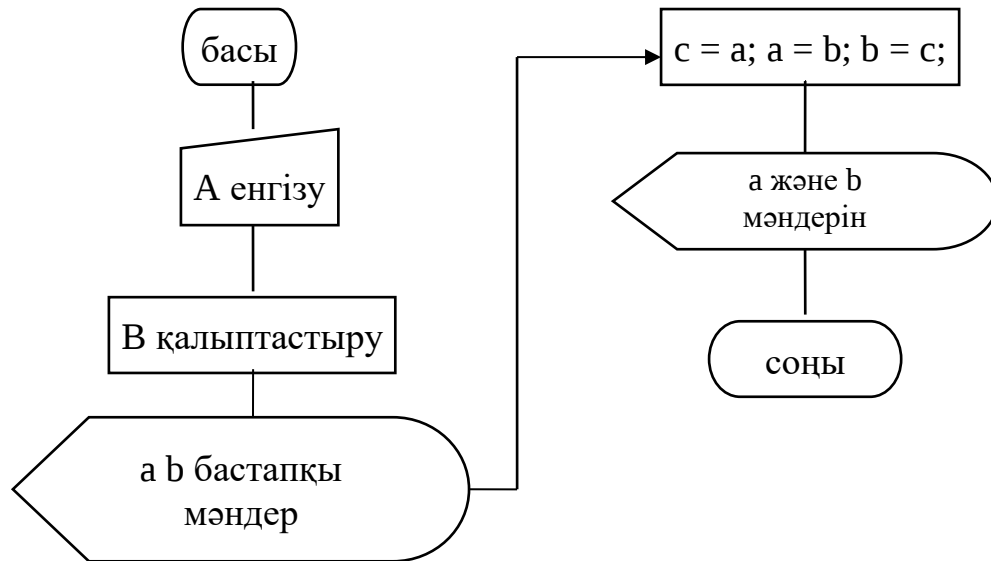
3 терезеде Program.cs редакторының бетінде есепті шешудің бағдарламалық коды жазылады.

Есепті шығару алгоритмі өте қарапайым:

- а айнымалының мәнін диалог режимінде енгізуді ұйымдастыру керек;
- кездейсоқ түрде берілген диапозонда b айнымалысының мәнін көрсету;
- айнымалылардың кіріс мәндерін шығару;
- айнымалылардың мәндерін алмастыру;
- а және b айнымалыларының жаңа мәндерін басу.

Көрсетілген алгоритмді орындау үшін алдыңғы екі дәріс материалын оқу керек.

Өздік жұмыстың келесі деңгейі есепті шешу алгоритімінің құрылымдық схемасын сызу.



1.4 сурет – Есепті шешу алгоритмінің құрылымдық схемасы

Есепті шешу алгоритмінің құрылымдық схемасын қолдана отырып, бағдарламалау кодын құраймыз:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace ConsoleApplication1
{
    class Program
    {
        static void Main()
        {
            int a, b, c;
            string buf;
            //Диалог режимінде а айнымалының мәнін енгізу
            Console.Write("a-нің бүтін мәнін енгізіңіз");
            buf = Console.ReadLine();
            a = Convert.ToInt32(buf);
            // b айнымалының мәнін кездейсоқ түрде қалыптастыру
            Random rnd = new Random();
            b = rnd.Next() % 21;
            //Бастапқы мәндерді баспаға шығару
            Console.WriteLine("a және b айнымалылардың бастапқы мәндері:");
            Console.WriteLine("a={0} b={1}", a, b);
            //Ауыстыру
            c = a; a = b; b = c;
            // Ауыстырудан кейінгі a және b айнымалылардың мәндерін баспаға шығару
        }
    }
}
  
```

```

Console.WriteLine("a және b айнымалылардың жаңа мәндері:");
Console.WriteLine("a={0} b={1}", a, b);
// Задержка рабочего экрана монитора
Console.WriteLine("Жалғастыру үшін Enter пернесін
басыңыз");
Console.ReadLine();
}
}
}

```

Бағдарлама жұмысы:

a-нің бүтін мәнін енгізіңіз 35

a және b айнымалылардың бастапқы мәндері:

a=35 b=13

a және b айнымалылардың жаңа мәндері:

a=13 b=35

Жалғастыру үшін Enter пернесін басыңыз

Зертханалық жұмыс бойынша есепхатты дайындау кезінде келесі құрылымды және реттілікті ұстану керек:

- титул беті;
- зертханалық жұмыстың атауы;
- зертханалық жұмыстың мақсаты;
- зертханалық жұмыстың жеке тапсырмасы;
- жеке тапсырма бойынша қысқаша түсініктемелер (керек болса есеп шығару алгоритмінің құрылымдық схемасы);
- жеке тапсырманың тиісті бағдарламалық код;
- бағдарлама жұмысының нәтижесі;
- қорытынды.

Титул беті есеп-хаттың бірініші беті және ақпарат көзі болып табылады , құжатты іздеу үшін қажет. Сондықтан , ол министрлік атауын , университеттің атауын , бөлім атауын , пәннің атауы, модуль атауын , зертханалық жұмысты орындаған студент аты-жөнін, және зертханалық жұмысты қабылдаған оқытушының аты-жөні көрсетіледі. Титул бетінің төменгі жағында қай жерде және қашан орындалғаны көрсетіледі, мысалы, Өскемен 2010 жыл.

Титул беті есеп-хаттың жалпы бет санына кіреді, бірақ титул бетінде беттің нөмірі қойылмайды.

Зертханалық жұмыс атауы зертханалық жұмысты орындауға арналған нұсқаулық ішіндегі атауымен сәйкес келуі керек, ол жиі модуль атауымен сәйкес келеді.

Зертханалық жұмыс мақсаты пән модулі мақсатының қысқаша сипаттамасы болып табылады.

Зертханалық жұмыстағы жеке тапсырма жеке тапсырманың толық мәтінін құрайды, зертханалық жұмыс бойынша үй тапсырмасын орындағаннан кейін ғана мұғалімнен алады.

Жеке тапсырмалар бойынша қысқаша түсініктемелер жеке тапсырмаларды орындау алгоритмін сипаттайды. Егер қажет

болса, алгоритімнің құрылымдық схемасы немесе егжей-тегжейлі ауызша сипаттамасы қолданылады.

Жеке жұмысты орындауға қажетті код бағдарлама кодының толық мәтінін құрайды, немесе бағдарлама кодының үзіндісін, бағдарламалау ортасына арналған студент өзі бағдарламалаған, онсыз орындаған жұмысты түсіндіру мүмкін емес.

Бағдарламаның нәтижелері әдетте оның барлық режимдерде бағдарлама терезесінің көшірмелерін қамтиды .

Алынған нәтижелер , әдетте, зертханалық жұмыс нәтижесін көрсетеді.

Есеп-хат мәтінінің беттері А4 пішіміне сәйкес болуы керек.

Есеп-хатты басып шығару машиналық жолмен немесе компьютер құрылымын пайдаланып басып шығару немесе графикалық жолмен ақ беттің бір жағына басып шығару керек. Қаріп түрі - TimesNewRoman, негізгі қаріп өлшемі - № 14, рұқсат етіледі № 12. Негізгі интервал -1, рұқсат етіледі - 1,5.

Есеп-хат мәтінін келесідей басу керек: сол, жоғарғы, төменгі және оң – 20 мм.

Шегінісі кем дегенде төрт ондық басталады және бір құжат аясында бірдей болуы керек.

Есеп-хатты орындау тәсіліне қарамастан, мәтіннің басылғаны, суреттер, кестелер және қосымшалар сапасы айқын талаптарға сәйкес болуы керек.

Есеп-хатта белгілі бір сөздер, формулалар тек қара сиямен жазылуы тиіс, жазылған мәтін тығыздығы негізгі мәтіннің тығыздығына жақын болуы тиіс.

Қателіктер мен графикалық дәлсіздіктер қара сиямен, суретті машинамен немесе қолмен ақ бояумен тазалануы керек.

Есеп-хат беттері зақымданған, дақтар және белгілер толық жойылуға рұқсат етілмейді.

Есеп-хаттағы бөлімдер нүктесіз араб цифрларымен нөмірленген және абзацтан жазылған болуы тиіс. Бөлімшелер әрбір бөлімге сәйкес нөмірленуі тиіс. Бөлім нөмері саны бөлінген бөлім мен бөлімшелерден тұрады. Қосалқы нөмер аяғында нүкте қойылмайды. Бөлімшелердің ретінде бөлімдер бір немесе бірнеше элементтерден тұруы мүмкін.

Құжатта бөлім атаулары жолдың ортасында нүктесіз орналастырылады және бас әріппен астын сызбай және арнайы белгілемей жазылады.

Құжаттың бөлімшелер атауы жолдың ортасында кіші әріппен, бас әріппен бастап қою қаріппен жазылады.

Бөлімнің қосалқы бөлімдерді жоқ болса , онда элементтерді нөмірлеу сол бөлімде шегінде болуы тиіс, және элемент саны нүктемен бөлінген бөлім нөмірінен және элементтен тұруы тиіс .

Бөлімнің қосалқы бөлімдер бар болса , тармақтарды нөмірлеу әрбір бөлімшені және элемент саны шегінде болуы тиіс сандардан тұрады , тармақ және нүктелермен бөлінген болуы тиіс.

Бүкіл мәтінде нөмірлеуді сақтай отырып, есеп-хатты араб сандайрмен нөмірлеу керек.

Есеп-хаттың беттер нөмері беттің жоғарғы жағында ортасында соңында нүктесіз жазылуы керек.

Есеп-хат беттері қыстырғышпен қыстырылады немесе файлға салынады (степпер қолданбаңыз).

1.4 Зертханалық жұмыс бойынша үй тапсырмасы

Кесіндінің ұзындығын есептеу бағдарламасын жазу керек. Кесінді ұштарының координаттары - $A(x_1, y_1)$ және $B(x_2, y_2)$. А нүктесі координаттарының мәні диалог режимінде енгізіледі, ал В нүктесін минус 100-ден плюс 100-ге дейінгі аралықта кездейсоқ түрде беріледі. Бағдарлама жұмысының нәтижесін монитор экранына шығаруды қарастыру керек.

1.5 СӨЖ арналған жеке тапсырмалары

1.5.1 Өрнекті есептейтін программаны жазыңыз:

$$Y=3*(X+1)^3+4*(X-1)+2.$$

X мәні диалог режимінде енгізіледі. Бір-бірден қарастыру әдісін қолданып (методом перебора) Y нөлге тең болатындай X мәнін табыңыз. Бағдарлама жұмысының нәтижесін монитор экранына шығаруды қарастыру керек.

1.5.2 Қосындыны есептейтін программаны жазыңыз:

$$S=1+1/2+1/3+1/4+1/5.$$

Бағдарлама жұмысының нәтижесін монитор экранына шығаруды қарастыру керек.

1.5.3 А және В айнымалыларының мәндерін алмастыруды орындайтын алгоритмі бар программаны жазыңыз (қосымша с айнымалысы қолданылмауы тиіс). Айнымалы мәндерін диалог режимінде енгізіледі. Бағдарлама жұмысының нәтижесін монитор экранына шығаруды қарастыру керек.

1.5.4 Диалог режимінде енгізілген нақты санның бүтін және бөлшек бөліктерін бөлетін программаны жазыңыз және оларды монитор экранына бөлек шығарыңыз.

1.5.5 Үшбұрыштың ауданын есептейтін программаны жазыңыз, үшбұрыштың А, В, С жақтары диалог режимінде енгізіледі.

1.5.6 Үш кесіндіден тұратын сынық кесінді ұзындығының есептейтін программаны жазыңыз. Кесінді ұштарының координаттары $A(x,y)$, $B(x,y)$, $C(x,y)$ және $D(x,y)$. Координаттардың мәндері 0-ден 100 дейінгі аралықтан кездейсоқ түрде алынады. Бағдарлама жұмысының нәтижесін монитор экранына шығаруды қарастыру керек.

1.5.7 Өрнекті есептейтін программаны жазыңыз: $C = A^2 + \sqrt{(A + \ln B) / 2}$. А және В айнымалыларының мәндері диалог режимінде енгізіледі.

1.5.8 Шеңбердің ауданын есептейтін программаны жазыңыз. Радиустың мәні диалог режимінде енгізіледі және ол нөлден үлкен болуы тиіс.

1.5.9 Кесіндінің ұзындығы дюймде берілген. (1 дюйм = 2,54 см.), диалог режимінде енгізіледі. Ұзындықтың мәнін метрлік жүйеге, яғни метрге, сантиметрге және миллиметрге айналдырыңыз. Мысалы, 21 дюйм = 0м 53см 3,4мм.

1.5.10 α бұрышы диалог режимінде градуспен, минутпен және секундпен беріледі. Оның мәнін радиан шамасында табыңыз.

1.5.11 Екі нақты санның мәні диалог режимінде беріледі. Значения двух действительных чисел задается в режиме диалога. Найти среднее арифметическое этих чисел и среднее геометрическое их модулей.

1.5.12 t_1 температурасымен v_1 литр су t_2 температурасымен v_2 литр су араластырылған. Алынған қоспаның температурасын және көлемін тап. v_1 , t_1 , v_2 және t_2 диалог режимінде беріледі.

1.5.13 Шеңбердің ұзындығы диалог режимінде беріледі. Шеңбермен шектелген дөңгелектің ауданын тап.

1.5.14 Қабырғалары тең үшбұрыштың ұзындығы диалог режимінде енгізіледі. Бұл үшбұрыштың ауданын табу керек.

1.5.15 Үшбұрыш төбелерінің координаталары диалог режимінде берілген. Бұл үшбұрыштың периметрін есепте.

1.5.16 Куб қабырғасының ұзындығы диалог режимінде енгізіледі, Кубтың көлемін тап, және оның барлық беттерінің ауданын тап.

1.5.17 Екі кедергінің мәні диалог режимінде беріледі. Олардың тізбектей және параллель қосылу кезіндегі кедергіні тап.

1.5.18 X мәні диалог режимінде енгізіледі. Тек қана арифметикалық операцияларды жүргізе отырып, өрнекті есепте

$$Y = 1 - 2 * X + 3 * X^2 - 4 * X^3.$$

Бір өрнекте операциялардың саны сегізден аспау керек.

1.5.19 X мәні диалог режимінде енгізіледі. Тек қана арифметикалық операцияларды жүргізе отырып, өрнекті есепте

$$Y = 1 + 2 * X + 3 * X^2 + 4 * X^3.$$

Бір өрнекте операциялардың саны сегізден аспау керек.

1.5.20 Сақинаның ауданын тап, оның сыртқы радиусы 100 тең, ал ішкі радиусы диалог режимінде беріледі және 100-ден аспау керек.

1.6 СОӨЖ бойынша бақылау сұрақтары

1.6.1 NET платформа ұғымы дегеніміз не?

1.6.2 Тілдік орта ұғымы нені білдіреді? (CommonLanguageRuntime, CLR)?

1.6.3 «қосымша» термині нені білдіреді?

1.6.4 «жоба» термині нені білдіреді? Оның құрамы?

1.6.5 «Есім кеңістігі» нені қамтиды?

1.6.6 Visual Studio.NET ортасы нені қамтиды?

1.6.7 Алгоритм түсінігі? Алгоритмнің негізгі қасиеті.

1.6.8 Есепті шешу қадамдары?