**Лабораторная работа №3**

**Разработка программ с использованием операторов цикла**

**Цель работы:**

Познакомиться с операторами цикла, научиться использовать их в программах с использованием среды **Visual Studio.Net.**

**1. Теоретические сведения**

Операторы цикла используются для организации многократно повторяющихся вычислений. К операторам цикла относятся: цикл с предусловием while, цикл с постусловием do while, цикл с параметром for и цикл перебора foreach.

***Цикл с предусловием while***

Оператор цикла while организует выполнение одного оператора (простого или составного) неизвестное заранее число раз. Формат цикла while:

while (B) S;

где B – выражение, истинность которого проверяется (условие завершения цикла);

S – тело цикла (простой или составной оператор).

Перед каждым выполнением тела цикла анализируется значение выражения В: если оно истинно, то выполняется тело цикла, и управление передается на повторную проверку условия В; если значение В ложно – цикл завершается и управление передается на оператор, следующий за оператором S.

Если результат выражения B окажется ложным при первой проверке, то тело цикла не выполнится ни разу. Отметим, что если внутри цикла не будет оператора (или операторов), в результате выполнения которых условие B на какой-то итерации станет ложным, то произойдет зацикливание, то есть невозможность выхода из цикла. Поэтому внутри тела должны находиться операторы, приводящие к изменению значения выражения B так, чтобы цикл мог корректно завершиться.

В качестве иллюстрации выполнения цикла while рассмотрим программу вывода на экран целых чисел из интервала от 1 до n.

static void Main()

{

Console.Write("N= ");

int n=int.Parse(Console.ReadLine());

int i = 1;

while (i <= n) //пока i меньше или равно n

{

//выводим i на экран, затем увеличиваем его на 1

Console.Write(" {0}", i++ );

}

}

***Цикл с постусловием do while***

Оператор цикла do while также организует выполнение одного оператора (простого или составного) неизвестное заранее число раз. Однако в отличие от цикла while условие завершения цикла проверяется после выполнения тела цикла. Формат цикла do while:

do S while (B);

где В – выражение, истинность которого проверяется (условие завершения цикла);

S – тело цикла (простой или составной оператор).

Сначала выполняется оператор S, а затем анализируется значение выражения В: если оно истинно, то управление передается оператору S, если ложно - цикл завершается, и управление передается на оператор, следующий за условием B. Так как условие В проверяется после выполнения тела цикла, то в любом случае тело цикла выполнится хотя бы один раз. В операторе do while, так же как и в операторе while, возможна ситуация зацикливания в случае, если условие В всегда будет оставаться истинным.

В качестве иллюстрации выполнения цикла do while рассмотрим программу вывода на экран целых чисел из интервала от 1 до n.

static void Main()

{

Console.Write("N= ");

int n=int.Parse(Console.ReadLine());

int i = 1;

do

Console.Write(" {0}", i++); //выводим i на экран, затем увеличиваем его на 1

while (i <= n); //пока i меньше или равно n

}

***Цикл с параметром for***

Цикл с параметром имеет следующую структуру:

for ( <инициализация>; <выражение>; <модификация>) <оператор>;

*Инициализация* используется для объявления и/или присвоения начального значения величине, используемой в цикле в качестве параметра (счетчика). Областью действия переменной, объявленной в части инициализации цикла, является цикл и вложенные блоки. Инициализация выполняется один раз в начале исполнения цикла.

*Выражение* определяет условие выполнения цикла: если его результатом является истина, цикл выполняется. Истинность выражения проверяется перед каждым выполнением тела цикла, таким образом, цикл с параметром реализован как цикл с предусловием.

*Модификация* выполняется после каждой итерации цикла и служит обычно для изменения параметра цикла.

*Оператор* (простой или составной) представляет собой тело цикла. Любая из частей оператора for (инициализация, выражение, модификация, оператор) может отсутствовать, но точку с запятой, определяющую позицию пропускаемой части, надо оставить.

static void Main()

{

Console.Write("N= ");

int n=int.Parse(Console.ReadLine());

for (int i=1; i<=n; i++)

{

Console.Write(" {0} ,}", i);

}

}

***Вложенные циклы***

Циклы могут быть простые или вложенные (кратные, циклы в цикле).

Вложенными могут быть циклы любых типов: while, do while, for. Каждый внутренний цикл должен быть полностью вложен во все внешние циклы. «Пересечения» циклов не допускаются.

Рассмотрим пример использования вложенных циклов, который позволит вывести на экран числа следующим образом:

2 2 2 2 2

2 2 2 2 2

2 2 2 2 2

2 2 2 2 2

static void Main()

{

for (int i = 1; i <= 4; ++i)

{

for (int j=1; j<=5; ++j)

{

Console.Write(”2 ”);

}

Console.WreLine();

}

}

***Задания.***

1. Измените программу так, чтобы таблица содержала n строк и m столбцов (значения n и m вводятся с клавиатуры).

2. Измените программу так, чтобы на экран выводилась прямоугольная таблица, в которой n строк, m столбцов, а каждый элемент таблицы равнялся сумме номера строки и номера столбца.

**2. Примеры решения практических задач**

1. Написать программу, которая выводит на экран квадраты всех четных чисел из диапазона от А до В (А и В целые числа, при этом А≤В).

Указания по решению задачи. Из диапазона целых чисел от А до В необходимо выбрать только четные числа. Напомним, что четными называются числа, которые делятся на два без остатка. Кроме того, четные числа представляют собой упорядоченную последовательность, в которой каждое число отличается от предыдущего на 2. Решить эту задачу можно с помощью каждого оператора цикла.

using System;

namespace Example

{

class Program

{

static void Main()

Console.Write("a= ");

int a=int.Parse(Console.ReadLine());

Console.Write("b= ");

int b=int.Parse(Console.ReadLine());

int i;

Console.Write("FOR: ");

a=(a%2==0)? a: a+1;

for (i=a; i<=b; i+=2)

{

Console.Write(" {0}", i\*i);

}

Console.Write("\nWHILE: ");

i=a;

while (i<=b)

{

Console.Write(" {0}", i\*i);

i+=2;

Console.Write("\nDO: ");

i=a;

do

{

Console.Write(" {0}", i\*i);

i+=2;

}

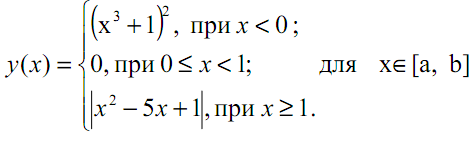
while (i<=b);

}

}

}

***Задание***. Измените программу так, чтобы она выводила на экран только нечетные числа из диапазона от А до В (А и В целые числа, при этом А≤В).

2. Постройте таблицу значений функции 

с шагом h.

using System;

namespace Example

{

class Program

{

static void Main()

{

Console.Write("a= ");

double a=double.Parse(Console.ReadLine());

Console.Write("b= ");

double b=double.Parse(Console.ReadLine());

Console.Write("h= ");

double h=double.Parse(Console.ReadLine());

double y;

int i=1;

Console.WriteLine("{0,3} {1,5} {1,5}","#","x", "f(x)");

for (double x=a; x<=b; x+=h, ++i)

{

if (x<0)

y=Math.Pow(Math.Pow(x,3)+1,2);

}

else

{

if (x<1)

{

y=0;

}

else

{

y=Math.Abs(x\*x-5\*x+1);

}

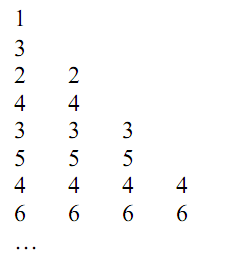
}

Console.WriteLine("{0,3} {1,5:f2} {2,5:f2}",i,x,y);

}

} } }

3. Напечатать числа в виде следующей таблицы:



using System;

namespace Example

{

class Program

{

static void Main()

{

Console.Write("n= ");

byte n=byte.Parse(Console.ReadLine());

for (byte i=1 ; i<=n; ++i)

{

for (byte j=1; j<=i; ++j)

{

Console.Write("{0,4}",i);

}

Console.WriteLine();

for (byte j=1; j<=i; ++j)

{

Console.Write("{0,4}",i+2);

}

Console.WriteLine();

}

}

}

}

**Задания для самостоятельной работы**

I. Вывести на экран:

*Замечание*. При решении задачи можно использовать любой из операторов цикла while, do while и for. Свой выбор обоснуйте

1) все целые числа из диапазона от А до В (А≤В), оканчивающиеся на цифру Х;

2) все целые числа из диапазона от А до В (А≤В), оканчивающиеся на цифру Х или У;

3) все целые числа из диапазона от А до В (А≤В), оканчивающиеся на любую четную

цифру;

4) только положительные целые числа из диапазона от А до В (А≤В);

5) все целые числа из диапазона от А до В, кратные трем (А≤В);

6) все четные числа из диапазона от А до В, кратные трем (А≤В);

7) только отрицательные четные числа из диапазона от А до В (А≤В);

8) все двухзначные числа, в записи которых все цифры разные;

9) все трехзначные числа, которые начинаются и заканчиваются на одну и ту же

цифру;

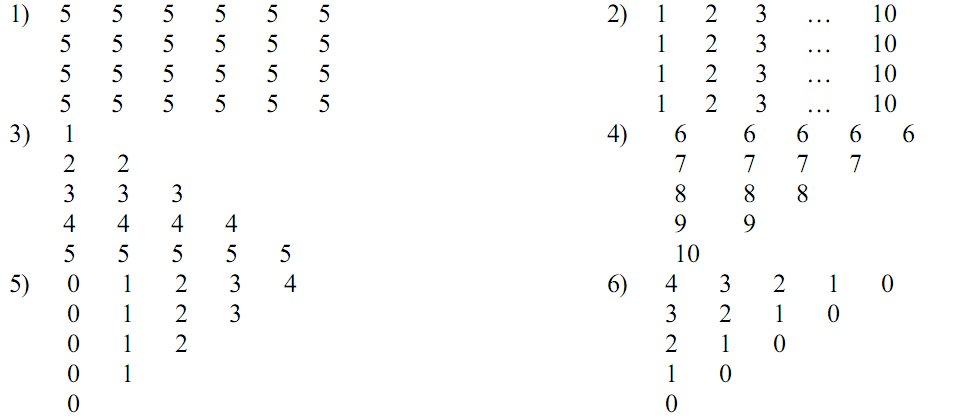
10) все трехзначные числа, в которых хотя бы две цифры повторяются;

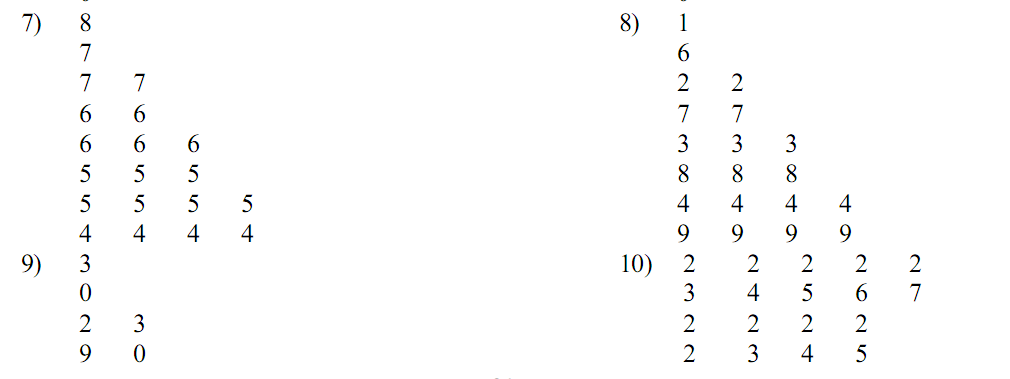
11) все целые числа из диапазона от А до В (А≤В), оканчивающиеся на любую нечетную

цифру;

12) только положительные четные целые числа из диапазона от А до В (А≤В)

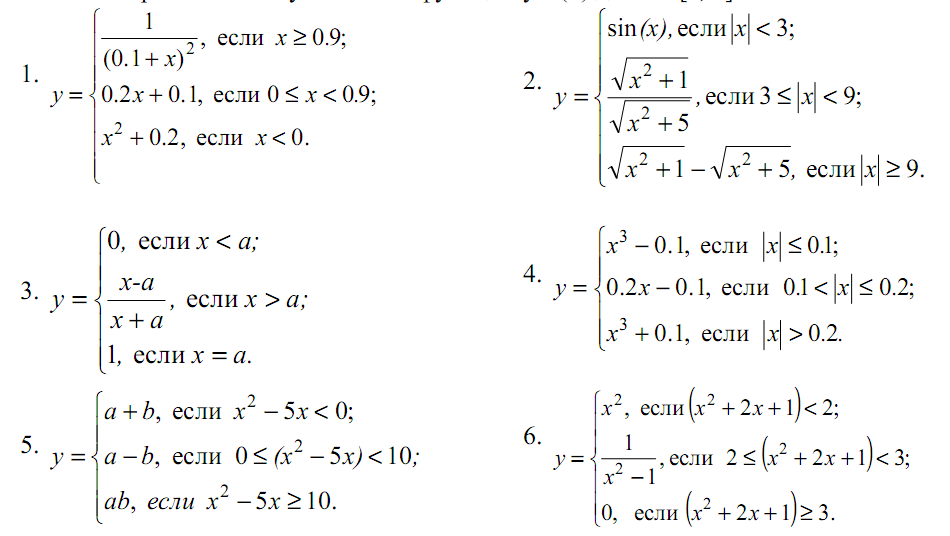
II. Вывести на экран числа следующим образом:

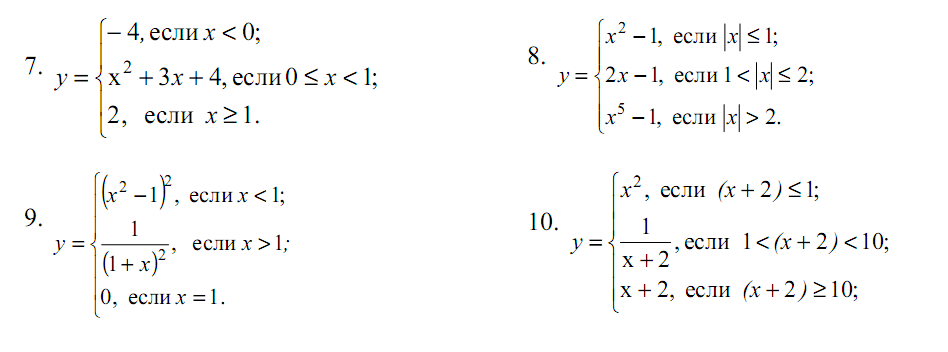






III. Постройте таблицу значений функции y=f(x) для х∈[a, b] с шагом h.





11



12. 