

### Теоретический материал

#### Циклы:

Циклы являются управляющими конструкциями, позволяя в зависимости от определенных условий выполнять некоторое действие множество раз. В C# имеются следующие виды циклов:

- **for**
- **foreach**
- **while**
- **do...while**

#### Цикл **for**

```
for([действия_до_выпол-ния_цикла];[условие];[действия_после_выпол-ния])  
  
    {  
  
        // действия  
  
    }
```

#### Цикл **do..while**

В цикле **do** сначала выполняется код цикла, а потом происходит проверка условия в инструкции **while**. И пока это условие истинно, цикл повторяется.

```
do

{

    //действия цикла

}

while (условие);
```

## Цикл **while**

Цикл **while** сразу проверяет истинность некоторого условия, и если условие истинно, то код цикла выполняется.

```
while (условие)

{

    //действия цикла

}
```

## Цикл **foreach**

Цикл **foreach** предназначен для перебора набора или коллекции элементов. Его общее определение:

```
foreach(тип_данных переменная in коллекция)

{

    // действия цикла

}
```

## Операторы **break** и **continue**

Иногда возникает ситуация, когда требуется выйти из цикла, не дожидаясь его завершения. В этом случае мы можем воспользоваться оператором **break**.

```
for (int i = 0; i < 9; i++)  
  
    {  
  
        if (i == 5)  
  
            break;  
  
        Console.WriteLine(i);  
  
    }
```

Если необходимо, чтобы при проверке цикл не завершался, а просто пропускал текущую итерацию. Для этого можно воспользоваться оператором **continue**:

```
for (int i = 0; i < 9; i++)  
  
    {  
  
        if (i == 5)  
  
            continue;  
  
        Console.WriteLine(i);  
  
    }
```

Одни циклы могут быть вложенными в другие:

```
for (int i = 1; i < 10; i++)
```

```
{  
  
    for (int j = 1; j < 10; j++)  
  
    {  
  
        Console.Write($"{i * j} \t");  
  
    }  
  
    Console.WriteLine();  
  
}
```

## Задание 1

### Задача:

#### Ряды

Реализуйте программный продукт средствами языка C# со следующим функционалом:

Вычисление значения функции  $f(x)$  (соответствующей вашему варианту) с помощью ряда Маклорена с заданной точностью  $\epsilon$  ( $\epsilon$  и  $x$  вводятся с клавиатуры,  $\epsilon < 0.01$ );

Вычисление  $n$ -го члена ряда ( $n$  и  $x$  вводятся с клавиатуры).

Вариант №	Ряд
1, 11, 21, 31.	$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!} + \dots, \forall x \in (-\infty; +\infty)$
2, 12, 22, 32.	$\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \dots + (-1)^{n+1} \frac{x^n}{n} + \dots, \forall x \in (-1; 1]$
3, 13, 23, 33.	$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!} + \dots, \forall x \in (-\infty; +\infty)$
4, 14, 24, 34.	$\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots + (-1)^n \frac{x^{2n}}{(2n)!} + \dots, \forall x \in (-\infty; +\infty)$
5, 15, 25, 35.	$\arcsin x = x + \frac{x^3}{2!! \cdot 3} + \frac{3!!x^5}{4!! \cdot 5} + \frac{5!!x^7}{6!! \cdot 7} + \dots + \frac{(2n-1)!!x^{2n+1}}{(2n)!!(2n+1)} + \dots, \forall x \in (-1; 1)$
6, 16, 26, 36.	$\arctg x = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots + (-1)^n \frac{x^{2n+1}}{2n+1} + \dots, \forall x \in (-1; 1)$
7, 17, 27, 37.	$\operatorname{sh} x = x + \frac{x^3}{3!} + \frac{x^5}{5!} + \frac{x^7}{7!} + \dots + \frac{x^{2n+1}}{(2n+1)!} + \dots, \forall x \in (-\infty; +\infty)$
8, 18, 28, 38.	$\operatorname{ch} x = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \frac{x^6}{6!} + \dots + \frac{x^{2n}}{(2n)!} + \dots, \forall x \in (-\infty; +\infty)$

9, 19, 29, 39.	$\operatorname{arsh} x = x - \frac{x^3}{2!! \cdot 3} + \frac{3!! x^5}{4!! \cdot 5} - \frac{5!! x^7}{6!! \cdot 7} + \dots + (-1)^n \frac{(2n-1)!! x^{2n+1}}{(2n)!! (2n+1)} + \dots, \forall x \in (-1; 1)$
10, 20, 30, 40.	$\operatorname{arch} x = x + \frac{x^3}{3} + \frac{x^5}{5} + \frac{x^7}{7} + \dots + \frac{x^{2n+1}}{2n+1} + \dots, \forall x \in (-1; 1)$

**Решение:**

```

C# Прокошин Кирилл ЭФБО 07 23
using System;

class Program
{
    static double CalculateCosine(double x, double epsilon)
    {
        double term = 1, sum = term;
        int n = 1;
        while (Math.Abs(term) >= epsilon)
        {
            term *= -x * x / (2 * n * (2 * n - 1));
            sum += term; n++;
        }
        return sum;
    }

    static void Main()
    {
        Console.WriteLine("input X:");
        double x = double.Parse(Console.ReadLine());
        Console.WriteLine("input E:");
        double e = double.Parse(Console.ReadLine());
        if (e < 0.01) Console.WriteLine(CalculateCosine(x, e));
    }
}

```

**Ответ:**

```

C# Консоль отладки Microsoft V
input X:
1
input E:
0,001
0,5403025793650793

```

## Задание 2

**Задача:**

## Счастливым билет

Если на билете сумма первых трёх цифр в номере билета равна сумме трёх последних, то этот билет считается счастливым. Напишите программу, которая получала бы на вход шестизначный номер билета и выводила, счастливый это билет или нет. К примеру: билеты 777 777 и 255 642 — счастливые, а 123 456 — нет.

Использовать при решении задачи можно только простые базовые типы (т.е. использование массивов, строк и коллекций запрещено, должно обрабатываться именно число).

### Пример выполнения:

\*\*\*\*\*

*Введите номер билета: 123060*

*True*

*Введите номер билета: 123040*

*False*

\*\*\*\*\*

Совет: Получайте разряды числа за счет использования операций «/» и «%».

## Решение:

```
Прокошин Кирилл ЭФБО 07 23
1 using System;
2
3 class Program
4 {
5
6     static void Main()
7     {
8         while (true)
9         {
10            Console.Write("Ticket number:");
11            int ticket = int.Parse(Console.ReadLine());
12            int sum1 = ticket / 100_000 + (ticket / 10_000) % 10 + (ticket / 1_000) % 10;
13            int sum2 = (ticket / 100) % 10 + (ticket / 10) % 10 + ticket % 10;
14            Console.WriteLine(sum1==sum2);
15        }
16    }
17 }
18
```

## Ответ:

```
Ticket number:123060
True
Ticket number:123040
False
Ticket number:
```

### Задание 3

#### Задача:

##### Сокращение дроби

Пользователь вводит числа М и N. Напишите программу, которая преобразует дробь М/Ν к несократимому виду и выдаёт получившийся результат.

##### **Пример выполнения:**

\*\*\*\*\*

*Введите числитель: 4*

*Введите знаменатель: 6*

*Результат: 2 / 3*

\*\*\*\*\*

*Введите числитель: 25*

*Введите знаменатель: 40*

*Результат: 5 / 8*

\*\*\*\*\*

*Введите числитель: -6*

*Введите знаменатель: 12*

*Результат: -1 / 2*

\*\*\*\*\*

*Введите числитель: 896*

*Введите знаменатель: 3584*

*Результат: 1 / 4*

\*\*\*\*\*

(При вводе отрицательных чисел в выводе должен быть только один ведущий минус. Неправильный вывод программы: 1/-2, -1/-2.).

Совет: знаменатель не может быть равен 0.

#### Решение:



```

Program.cs
Прокошин Кирилл ЭФБО 07 23
using System;

class Program
{
    static int GCD(int a, int b) => b == 0 ? Math.Abs(a) : GCD(b, a % b);

    static void Main()
    {
        while (true)
        {
            Console.Write("instert top number");
            int M = int.Parse(Console.ReadLine());
            Console.Write("instert bottom number");
            int N = int.Parse(Console.ReadLine());

            if (N == 0)
            {
                Console.Write("Zero exeption");
                return;
            }

            int gcd = GCD(M, N);
            M /= gcd;
            N /= gcd;

            if (N < 0) { M = -M; N = -N; }

            Console.WriteLine($"{M} / {N}");
        }
    }
}

```

Ответ:

```

instert top number4
instert bottom number6
2 / 3
instert top number25
instert bottom number40
5 / 8
instert top number-6
instert bottom number12
-1 / 2
instert top number896
instert bottom number3584
1 / 4
instert top number

```

#### Задание 4

Задача:

##### Угадай число

Напишите программу, которая угадывает число, задуманное пользователем. Число загадывается в диапазоне от 0 до 63. Программа задаёт вопросы вида «Ваше число больше такого-то?» и на основе

ответов пользователя («да-1» или «нет-0») угадывает число.

Алгоритм, должен давать ответ за семь вопросов.

### Решение:

```
Прокошин Кирилл ЭФБО 07 23
using System;

class Program
{
    static void Main()
    {
        int low = 0, high = 63;

        for (int i = 0; i < 6; i++)
        {
            int mid = (low + high) / 2;
            Console.WriteLine($"Ваше число больше {mid}? (да - 1, нет - 0)");
            int answer = int.Parse(Console.ReadLine());

            if (answer == 1)
                low = mid + 1;
            else
                high = mid;
        }

        Console.WriteLine($"Загаданное число: {low}");
    }
}
```

### Ответ:

```
Консоль отладки Microsoft Visual Studio
Ваше число больше 31? (да - 1, нет - 0)
1
Ваше число больше 47? (да - 1, нет - 0)
1
Ваше число больше 55? (да - 1, нет - 0)
1
Ваше число больше 59? (да - 1, нет - 0)
1
Ваше число больше 61? (да - 1, нет - 0)
1
Ваше число больше 62? (да - 1, нет - 0)
1
Загаданное число: 63
```

## Задание 5

### Задача:

## Кофейный аппарат

Кофейный аппарат может готовить два напитка: американо и латте. Для американо требуется 300 мл воды (цена 150 рублей), а для латте 30 мл воды и 270 мл молока (цена 170 рублей).

Напишите программу, которая спрашивает у пользователя (это действие программа делает один раз в начале работы), сколько всего миллилитров молока и воды залито в кофейный аппарат.

После чего начинается обслуживание пользователей, запрашивается, какой напиток хочет заказать посетитель. Пользователь выбирает один из двух напитков, программа отвечает одним из трёх вариантов: «Ваш напиток готов», «Не хватает воды» или «Не хватает молока», после чего переходит к обслуживанию следующего посетителя. Если молока и воды не хватает ни на один вид напитка, программа выдаёт отчёт и завершается.

В отчёте должно быть написано, что ингредиенты подошли к концу, должен быть указан остаток воды и молока в машине, должно быть указано, сколько всего было приготовлено чашек американо и латте за эту смену и итоговый заработок аппарата.

### **Пример выполнения:**

\*\*\*\*\*

*Введите количество воды в мл: 30*

*Введите количество молока в мл: 270*

*Выберите напиток (1 — американо, 2 — латте): 2*

*Ваш напиток готов.*

*\*Отчёт\**

*Ингредиентов осталось:*

*Вода: 0 мл*

*Молоко: 0 мл*

*Кружечек американо приготовлено: 0*

*Кружечек латте приготовлено: 1*

*Итого: 170 рублей.*

\*\*\*\*\*

**Решение:**

**Ответ:**

## Задание 6

**Задача:**

### Лабораторный опыт

В чашку Петри кладут  $N$  бактерий и добавляют  $X$  капель антибиотика ( $N$  и  $X$  вводятся с клавиатуры).

Известно, что число бактерий в чашке Петри увеличивается в два раза каждый час, а каждая капля антибиотика в первый час убивает 10 бактерий, во второй час — 9 бактерий, в следующий — 8 и так далее, пока антибиотик не перестанет действовать. Заметьте, что сначала число бактерий увеличивается, а затем только действует антибиотик.

Пользователь вашей программы вводит  $N$  и  $X$ , а программа печатает на экране, сколько бактерий останется в чашке Петри в конце каждого часа, до тех пор, пока не закончатся бактерии или антибиотик не перестанет действовать.

Цикл не должен быть бесконечным (после того как количество антибиотика или бактерий становится равным нулю выполнение программы должно быть завершено).

### **Пример выполнения:**

\*\*\*\*\*

*Введите количество бактерий: 12*

*Введите количество антибиотика: 1*

*После 1 часа бактерий осталось 14*

*После 2 часа бактерий осталось 19*

После 3 часа бактерий осталось 30

После 4 часа бактерий осталось 53

\*\*\*\*\*

### Решение:

```
Прокошин Кирилл ЭФБО 07 23 Program
1 using System;
2
3 class Program
4 {
5     static void Main()
6     {
7         Console.Write($"Бактерии:");
8         int N = int.Parse(Console.ReadLine());
9         Console.Write($"антибиотик");
10        int X = int.Parse(Console.ReadLine());
11
12        int hour = 1;
13        while (N > 0 && X > 0)
14        {
15            N *= 2;
16            int antibioticEffect = Math.Max(10 - (hour - 1), 0);
17            N -= antibioticEffect * X;
18            N = Math.Max(N, 0);
19            Console.WriteLine($"После {hour} часа бактерий осталось {N}");
20            hour++;
21            if (antibioticEffect == 0) break;
22        }
23    }
24 }
25
```

### Ответ:

```
Консоль отладки Microsoft Visual Studio
Бактерии:12
антибиотик1
После 1 часа бактерий осталось 14
После 2 часа бактерий осталось 19
После 3 часа бактерий осталось 30
После 4 часа бактерий осталось 53
После 5 часа бактерий осталось 100
После 6 часа бактерий осталось 195
После 7 часа бактерий осталось 386
После 8 часа бактерий осталось 769
После 9 часа бактерий осталось 1536
После 10 часа бактерий осталось 3071
После 11 часа бактерий осталось 6142
```

### Задание 7

#### Задача:

#### Колонизация Марса

Для освоения Марса требуется построить исследовательскую базу. База должна состоять из  $n$  одинаковых модулей, каждый из которых представляет собой прямоугольник.

Каждый модуль представляет собой жилой отсек, который имеет форму прямоугольника размером  $a \times b$  метров. Для повышения надежности модулей инженеры должны добавить вокруг каждого модуля слой дополнительной защиты. Толщина этого слоя должна составлять целое число метров, и все модули должны иметь одинаковую толщину дополнительной защиты.

Модуль с защитой, толщина которой равна  $d$  метрам, будет иметь форму прямоугольника размером  $(a + 2d) \times (b + 2d)$  метров.

Все модули должны быть расположены на заранее подготовленном прямоугольном поле размером  $w \times h$  метров. При этом они должны быть организованы в виде регулярной сетки: их стороны должны быть параллельны сторонам поля, и модули должны быть ориентированы одинаково.

Требуется написать программу, которая по заданным количеству  $n$  и размеру модулей  $a$  и  $b$ , а также размеру поля  $h$  и  $w$  для их размещения, определяет максимальную толщину слоя дополнительной защиты  $d$ , который можно добавить к каждому модулю. (Все данные вводятся с клавиатуры).

***Пример выполнения:***

\*\*\*\*\*

*Введите n: 11*

*Введите a: 2*

*Введите b: 2*

*Введите w: 21*

*Введите h: 25*

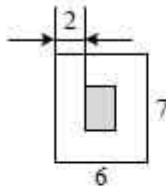
*Ответ d = 2*

\*\*\*\*\*

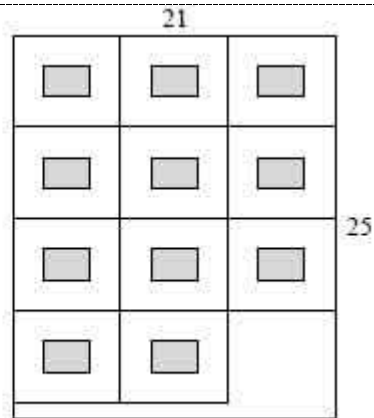
Для значений  $n = 11$ ,  $a = 2$ ,  $b = 3$ ,  $w = 21$ ,  $h = 25$ , возможный вариант реализации задачи представлен ниже.



Жилой  
модуль



Жилой модуль с  
дополнительной  
защитой  
толщиной 2



Размещение модулей на поле

## Решение:

```

1  using System;
2
3  Ссылка 0
4  class Program
5  {
6      Ссылка 1
7      static bool CanPlaceModules(int n, int a, int b, int w, int h, int d)
8      {
9          int newA = a + 2 * d;
10         int newB = b + 2 * d;
11         int horizontalFit = w / newA;
12         int verticalFit = h / newB;
13
14         if (horizontalFit * verticalFit >= n) return true;
15
16         horizontalFit = w / newB;
17         verticalFit = h / newA;
18
19         return horizontalFit * verticalFit >= n;
20     }
21
22     Ссылка 0
23     static void Main()
24     {
25         Console.WriteLine("n:");
26         int n = int.Parse(Console.ReadLine());
27         Console.WriteLine("a:");
28         int a = int.Parse(Console.ReadLine());
29         Console.WriteLine("b:");
30         int b = int.Parse(Console.ReadLine());
31         Console.WriteLine("w:");
32         int w = int.Parse(Console.ReadLine());
33         Console.WriteLine("h:");
34         int h = int.Parse(Console.ReadLine());
35
36         int left = 0, right = Math.Min(w, h);
37         while (left < right)
38         {
39             int mid = (left + right + 1) / 2;
40             if (CanPlaceModules(n, a, b, w, h, mid))
41                 left = mid;
42             else
43                 right = mid - 1;
44         }
45
46         Console.WriteLine($"Ответ d = {left}");
47     }
48 }

```

## Ответ:

