

## ТЕМА: ВВЕДЕНИЕ В ЯЗЫК ПРОГРАММИРОВАНИЯ «C++»

### Домашнее задание 2

#### ЗАДАНИЕ 1

Пользователь вводит число. Определить количество цифр в этом числе, посчитать их сумму и среднее арифметическое. Определить количество нулей в этом числе. Общение с пользователем организовать через меню.

Подсказка 1

Если у Вас возникли сложности с выполнением данного задания, нажмите кнопку «Подсказка». С полным решением задания вы сможете ознакомиться, нажав на кнопку «Решение»

Решение 1

#### ЗАДАНИЕ 2

Написать программу, которая выводит на экран шахматную доску с заданным размером клеточки.

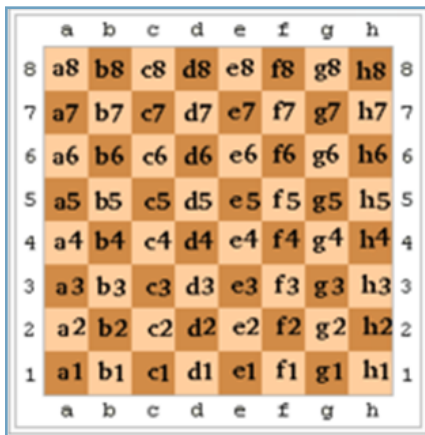


Рисунок 1

Например,

```

***_**_***_**_***_**_
***_**_***_**_***_**_
***_**_***_**_***_**_
_**_***_**_***_**_***
_**_***_**_***_**_***
_**_***_**_***_**_***

```

Подсказка 2

Если у Вас возникли сложности с выполнением данного задания, нажмите кнопку «Подсказка». С полным решением задания вы сможете ознакомиться, нажав на кнопку «Решение»

Решение 2

## ЗАДАНИЕ 3

Реализовать программу расчета заказа в кафетерии при условии, что заказ может быть на несколько человек и каждый клиент формирует свою часть заказа. Необходимо спросить у пользователя на сколько человек заказ. Далее каждому человеку выводиться меню (названия напитков, кондитерских изделий и их цена) и он выбирает. Предусмотреть возможность выбора нескольких элементов меню для клиента, если он желает добавить еще что-то в свой заказ. Результат работы программы — итоговая сумма общего заказа всей компании.

## ЗАДАНИЕ 4

Есть пустое поле игры «Морской бой».

	A	B	C	D	E	F	G	H	I	J
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										

Рисунок 2

Вывести в консоль номера полей игры:

A0 B0 C0 D0 E0 F0 G0 H0 I0 J0

A1 B1 C1 D1 E1 F1 G1 H1 I1 J1

.....

A9 B9 C9 D9 E9 F9 G9 H9 I9 J9

Финальные задания рассчитаны на самостоятельное решение. Здесь не будет подсказок и готового алгоритма. Вам необходимо применить все практические навыки, полученные из предыдущих заданий.

**ПОДСКАЗКА К ЗАДАНИЮ 1**

1. Вычисление суммы чисел в заданном диапазоне происходит путем накопления, т. е. на каждом шаге цикла к текущей сумме добавляется новое (следующее) число из диапазона.
2. Будет использоваться цикл с предусловием или постусловием?
3. Какое будет условие выполнения цикла?
4. Какое действие (действия) будут располагаться в теле цикла?
5. Следует ли помещать в тело цикла операцию вывода суммы в консоль?

## ПОДСКАЗКА К ЗАДАНИЮ 2

1. Традиционная шахматная доска представляет собой поле  $8 \times 8$  (всего 64) чередующихся по вертикали и горизонтали тёмных и светлых полей (клеток), т.е. 8 строк, внутри которых 8 ячеек.
2. Белая клетка формируется из символов «\*», а черная — из символов «-».
3. Строки доски выводятся сверху вниз, т.е. на первом шаге выводится восьмая (верхняя) строка, а на последнем — первая (нижняя) строка.
4. Каким образом связаны циклы формирования строки и ячейки шахматной доски?
5. Строки, которые выводятся (сверху вниз) на нечетном шаге цикла (первая сверху — 8 на примере доски, третья сверху — 6 на примере доски и т.д.) начинаются с белой клетки, т. е нечетные ячейки (a, c, e, g) в них формируются символом «\*», а четные (b, d, f, h) — символом «-», в строках, которые выводятся на четном шаге цикла (вторая сверху — 7 на примере доски и т.д.) — принцип обратный.
6. Каким образом можно проверить является ли строка (ячейка) четной или нечетной?
7. Одна строка шахматной доски формируется таким количеством линий, какой размер ячейки нужно создать, а внутри линии одинаковые символы (для формирования ячейки) также повторяются такое же количество раз.

## РЕШЕНИЕ ЗАДАНИЯ 1

### Описание решения

Вначале реализуем цикл для повторения вывода меню. Это будет цикл с постусловием, так как вначале необходимо хотя бы раз отобразить меню, а потом проверять, следует ли его повторять. Повторение вывода меню происходит, если пользователь ввел непредусмотренный элемент меню или после выполнения одного из пунктов меню. Для остановки программы следует предусмотреть пункт «Выход».

Меню будет содержать пять пунктов: определить количество цифр в числе, посчитать сумму цифр числа, посчитать среднее арифметическое цифр числа, определить количество нулей в числе и пятый пункт меню — выход.

Далее необходимо организовать процесс извлечения последней цифры текущего числа в цикле.

На первом шаге текущее число — это оригинальное число, заданное пользователем. Для извлечения последней цифры применяем операцию остатка от деления на 10, результат — последняя цифра.

После извлечения цифры нужно увеличить сумму (начальное значение которой ноль) на выделенную цифру.

Теперь выполним проверку, является ли извлеченная цифра нулем. Если да, то увеличиваем счетчик нулей на единицу.

Также необходимо увеличить счетчик цифр (количество цифр в числе) на единицу.

Следующее (подлежащее дальнейшей обработке) число — это начальная часть текущего числа без отделенной последней цифры.

Для получения оставшейся части числа выполняем операцию целочисленного деления на 10.

Т.к. пользователь может ввести любое N-значное число, то количество повторений цикла неизвестно. Значит, необходимо использовать тип цикла с предусловием (пока существует число, из которого можно выделять цифры, т.е. пока обрабатываемое число больше 0).

После завершения цикла нам известна сумма цифр числа и их количество. На основе этих данных мы можем вычислить среднее арифметическое, разделив сумму чисел на их количество.

Рассмотрим процесс на примере числа 238.

### Шаг 1.

Сумма равна изначально нулю, количество цифр — также.

На первом шаге к числу применяем операцию остатка от деления на 10, результат — последняя (третья) цифра:

$$238 \% 10 = 8 \text{ (не 0);}$$

$$\text{Сумма} = 0 + 8 = 8;$$

$$\text{Количество} = 0 + 1 = 1;$$

$$\text{Количество нулей} = 0.$$

Следующее (подлежащее дальнейшей обработке) число — это начальная часть текущего числа без отделенной последней цифры.

Для получения оставшейся части числа (первых двух цифр в нашем случае) выполняем операцию целочисленного деления на 10:

$$238 / 10 = 23.$$

Т.е. на следующем этапе (следующий шаг цикла) будет обрабатываться число 23.

**Шаг 2.**

$$23 \% 10 = 3 \text{ (не 0).}$$

$$\text{Сумма} = 8 + 3 = 11;$$

$$\text{Количество} = 1 + 1 = 2;$$

$$\text{Количество нулей} = 0;$$

$$\text{Оставшаяся часть числа} = 23 / 10 = 2.$$

**Шаг 3.**

$$2 \% 10 = 2 \text{ (не 0).}$$

$$\text{Сумма} = 11 + 2 = 13;$$

$$\text{Количество} = 2 + 1 = 3;$$

$$\text{Количество нулей} = 0;$$

$$\text{Оставшаяся часть числа} = 2 / 10 = 0. \text{ Цикл завершается.}$$

$$\text{Сумма} = 11 + 2 = 13;$$

$$\text{Количество} = 2 + 1 = 3;$$

$$\text{Количество нулей} = 0;$$

$$\begin{aligned} \text{Среднее арифметическое} &= \text{Сумма} / \text{Количество} \\ &= 13 / 3 = 4.33. \end{aligned}$$

1. Выводим строку в консоль с описанием названия программы (по желанию, данная строка кода может быть пропущена), создаем пять переменных целочисленного типа для хранения числа пользователя, выделенной цифры, номера пункта меню, количества цифр в числе, количества нулей в числе и две вещественных переменных для хранения суммы и среднего арифметического цифр числа.



Выводим в консоль строку — запрос с просьбой ввести целое число.

Считываем введенное пользователем число в соответствующую переменную.

Вначале реализуем цикл для повторения вывода меню. Это будет цикл с постусловием, так как вначале необходимо хотя бы раз отобразить меню, а потом проверять, следует ли его повторять. Повторение вывода меню происходит, если пользователь ввел непредусмотренный элемент меню или после выполнения одного из пунктов меню. Для останова программы следует предусмотреть пункт «Выход». Меню будет содержать пять пунктов: определить количество цифр в числе, посчитать сумму цифр числа, посчитать среднее арифметическое цифр числа, определить количество нулей в числе и пятый пункт меню — выход.

```
#include <iostream>
using namespace std;

int main()
{
    cout << "Home task #7.2.1\n";
    int number, digit, userChoice, nDigits, nZeros;
    float sum, mean;
    sum = 0;
    nDigits = 0;
    nZeros = 0;

    cout << "Please, enter your number:\n";
    cin >> number;
    do {
```

```
cout << "Your choice:\n";
cout << "1 - number of digits\n";
cout << "2 - sum of digits\n";
cout << "3 - mean\n";
cout << "4 - number of zeros\n";
cout << "5 - quit\n";
cin >> userChoice;
switch (userChoice) {
    case 1:
    {
        cout << "Calculating the number
                of digits...\n";
        break;
    }
    case 2:
    {
        cout << "Calculating the sum
                of digits...\n";
        break;
    }
    case 3:
    {
        cout << "Calculating the mean...\n";
        break;
    }
    case 4:
    {
        cout << "Calculating the number
                of zeros...\n";
        break;
    }
    case 5:
    {
```

```
        cout << "See you!";  
        break;  
    }  
    default:  
        cout << "Wrong menu item!";  
    }  
} while (userChoice != 5);  
  
return 0;  
}
```

2. Далее необходимо организовать процесс извлечения последней цифры текущего числа в цикле. Для извлечения последней цифры применяем операцию остатка от деления на 10, результат — последняя цифра.

После извлечения цифры нужно увеличить сумму (начальное значение которой ноль) на выделенную цифру. Теперь выполним проверку, является ли извлеченная цифра нулем. Если да, то увеличиваем счетчик нулей на единицу. Также необходимо увеличить счетчик цифр (количество цифр в числе) на единицу.

Для получения оставшейся части числа выполняем операцию целочисленного деления на 10.

```
#include <iostream>  
  
using namespace std;  
  
int main()  
{  
    cout << "Home task #7.2.1\n";
```

```
int number, digit, userChoice, nDigits, nZeros;
float sum, mean;
sum = 0;
nDigits = 0;
nZeros = 0;
cout << "Please, enter your number:\n";
cin >> number;

do {
    cout << "Your choice:\n";
    cout << "1 - number of digits\n";
    cout << "2 - sum of digits\n";
    cout << "3 - mean\n";
    cout << "4 - number of zeros\n";
    cout << "5 - quit\n";
    cin >> userChoice;

    while (number > 0)
    {
        digit = number % 10;
        sum += digit;
        nDigits++;
        if (digit == 0)
        {
            nZeros++;
        }
        number = number / 10;
    }
    mean = sum / nDigits;
    switch (userChoice) {
        case 1:
        {
            cout << "Calculating the number
                    of digits...\n";
```

```
        break;
    }
    case 2:
    {
        cout << "Calculating the sum
                of digits...\n";
        break;
    }
    case 3:
    {
        cout << "Calculating the mean...\n";
        break;
    }
    case 4:
    {
        cout << "Calculating the number
                of zeros...\n";
        break;
        break;
    }
    case 5:
    {
        cout << "See you!";
        break;
    }

    default:
        cout << "Wrong menu item!";
    }
} while (userChoice != 5);

return 0;
}
```

3. Добавляем в каждый кейсы вывод соответствующего результата в консоль.

```
#include <iostream>

using namespace std;

int main()
{
    cout << "Home task #7.2.1\n";

    int number, digit, userChoice, nDigits, nZeros;
    float sum, mean;
    sum = 0;
    nDigits = 0;
    nZeros = 0;

    cout << "Please, enter your number:\n";
    cin >> number;
    do {
        cout << "Your choice:\n";
        cout << "1 - number of digits\n";
        cout << "2 - sum of digits\n";
        cout << "3 - mean\n";
        cout << "4 - number of zeros\n";
        cout << "5 - quit\n";
        cin >> userChoice;

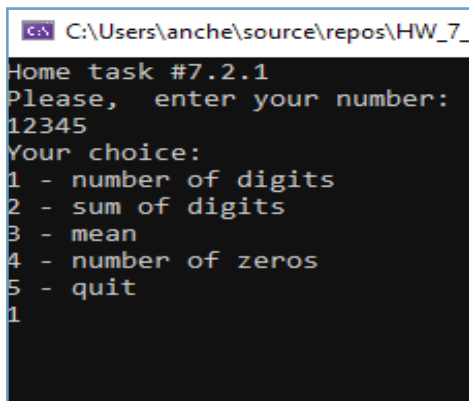
        while (number > 0)
        {
            digit = number % 10;
            sum += digit;
            nDigits++;
            if (digit == 0)
```

```
        {
            nZeros++;
        }
        number = number / 10;
    }
    mean = sum / nDigits;
    switch (userChoice) {
        case 1:
        {
            cout << "Calculating the number
                    of digits...\n";
            cout << nDigits << "\n";
            break;
        }
        case 2:
        {
            cout << "Calculating the sum
                    of digits...\n";
            cout << sum << "\n";
            break;
        }
        case 3:
        {
            cout << "Calculating the mean...\n";
            cout << mean << "\n";
            break;
        }
        case 4:
        {
            cout << "Calculating the number
                    of zeros...\n";
            cout << nZeros << "\n";
            break;
        }
    }
```

```
    }  
    case 5:  
    {  
        cout << "See you!";  
        break;  
    }  
    default:  
        cout << "Wrong menu item!";  
    }  
  
    } while (userChoice != 5);  
  
    return 0;  
}
```

Результаты тестирования программы

- Тест 1 — пользователь выбрал пункт меню «Количество цифр в числе»:



```
C:\Users\anche\source\repos\HW_7_  
Home task #7.2.1  
Please, enter your number:  
12345  
Your choice:  
1 - number of digits  
2 - sum of digits  
3 - mean  
4 - number of zeros  
5 - quit  
1
```

Рисунок 3



- Тест 2 — пользователь выбрал пункт меню «Сумма цифр числа»

```
Your choice:
1 - number of digits
2 - sum of digits
3 - mean
4 - number of zeros
5 - quit
2
Calculating the sum of digits...
15
```

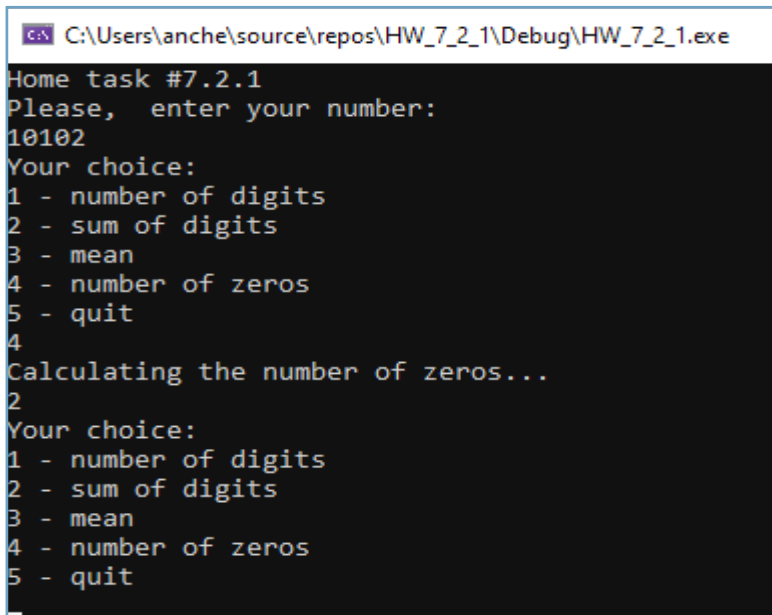
Рисунок 4

- Тест 3 — пользователь выбрал пункт меню «Среднее арифметическое цифр числа»

```
Your choice:
1 - number of digits
2 - sum of digits
3 - mean
4 - number of zeros
5 - quit
3
Calculating the mean...
3
.. ..
```

Рисунок 5

- Тест 4 — пользователь выбрал пункт меню «Количество нулей в числе»



```
C:\Users\anche\source\repos\HW_7_2_1\Debug\HW_7_2_1.exe
Home task #7.2.1
Please, enter your number:
10102
Your choice:
1 - number of digits
2 - sum of digits
3 - mean
4 - number of zeros
5 - quit
4
Calculating the number of zeros...
2
Your choice:
1 - number of digits
2 - sum of digits
3 - mean
4 - number of zeros
5 - quit
```

Рисунок 6

## РЕШЕНИЕ ЗАДАНИЯ 2

### Описание решения

Вначале реализуем цикл для формирования строк шахматной доски (длиной восемь, т.к. размер традиционной доски  $8 \times 8$ ).

В зависимости от того, является ли строка четной или нет, реализуем цикл (длиной заданного размера клетки) одного типа для формирования «черной» клетки или цикл второго типа для формирования «белой» клетки.

Внутри каждого из этих циклов нужно формировать цикл для создания ячеек доски (также длиной восемь).

Каждую ячейку проверяем на четность аналогично строке доски и в зависимости от этого реализуем цикл (длиной заданного размера клетки) одного типа для формирования фрагмента «черной» клетки (линии) или цикл второго типа для формирования фрагмента «белой» клетки.

### Решение

1. Выводим строку в консоль с описанием названия программы (по желанию, данная строка кода может быть пропущена), создаем одну переменную целочисленного типа для хранения размера ячейки.

Выводим в консоль строку — запрос с просьбой ввести размер ячейки.

Считываем введенное пользователем число в соответствующую переменную.

```
#include <iostream>

using namespace std;
int main()
{
    cout << "Home task #7.2.2\n";
    int cellSize;

    cout << "Input cell size:\n";
    cin >> cellSize;

    return 0;
}
```

2. Вначале реализуем цикл для формирования строк шахматной доски (длиной восемь, т.к. размер традиционной доски  $8 \times 8$ ).

Для каждой строки (т.е. на каждом шаге цикла) выполняем проверку, является ли строка четной или нет. Данную проверку реализуем с помощью операции получения остатка от деления на два. Если число делится на два без остатка (т. е. он равен нулю), то число — четное. Проверять будет индексы строк (т. е. значение переменной счетчика — цикла).

```
#include <iostream>

using namespace std;
int main()
{
```

```
cout << "Home task #7.2.2\n";
int cellSize;

cout << "Input cell size:\n";
cin >> cellSize;

for (int i = 0; i < 8; i++)
{
    if (i % 2 == 0)
    {
    }
}
else {
}
return 0;
}
```

3. В зависимости от того, является ли строка четной или нет, реализуем цикл (длиной заданного размера клетки) одного типа для формирования линий «черной» клетки или цикл второго типа для формирования линий «белой» клетки. Один шаг такого цикла выводит линию (часть строки доски) первого типа или второго типа. После вывода линии необходим переход на следующую строку консоли для вывода следующей линии.

```
#include <iostream>

using namespace std;
int main()
{
```

```
cout << "Home task #7.2.2\n";
int cellSize;
cout << "Input cell size:\n";
cin >> cellSize;
for (int i = 0; i < 8; i++)
{
    if (i % 2 == 0)
    {
        for (int m = 0; m < cellSize; m++)
        {
            cout << "\n";
        }
    }
    else {
        for (int m = 0; m < cellSize; m++)
        {
            cout << "\n";
        }
    }
    return 0;
}
```

4. Внутри цикла, реализованного на шаге 3, необходимо организовать цикл по формированию фрагментов ячеек доски (восемь ячеек внутри одной строки).

```
#include <iostream>

using namespace std;
int main()
{
```

```
cout << "Home task #7.2.2\n";
int cellSize;

cout << "Input cell size:\n";
cin >> cellSize;
for (int i = 0; i < 8; i++)
{
    if (i % 2 == 0)
    {
        for (int m = 0; m < cellSize; m++)
        {
            for (int j = 0; j < 8; j++)
            {
            }
            cout << "\n";
        }
    }
    else {
        for (int m = 0; m < cellSize; m++)
        {
            for (int j = 0; j < 8; j++)
            {
            }
            cout << "\n";
        }
    }
}
return 0;
}
```

5. Внутри цикла, реализованного на шаге 4, необходимо организовать проверку каждой ячейки на четность  $i$ , в зависимости от этого условия, реализовывать цикл для

вывода заданного (ширина ячейки в символах) количества символа «\*» или символа «-».

```
#include <iostream>

using namespace std;

int main()
{
    cout << "Home task #7.2.2\n";
    int cellSize;

    cout << "Input cell size:\n";
    cin >> cellSize;

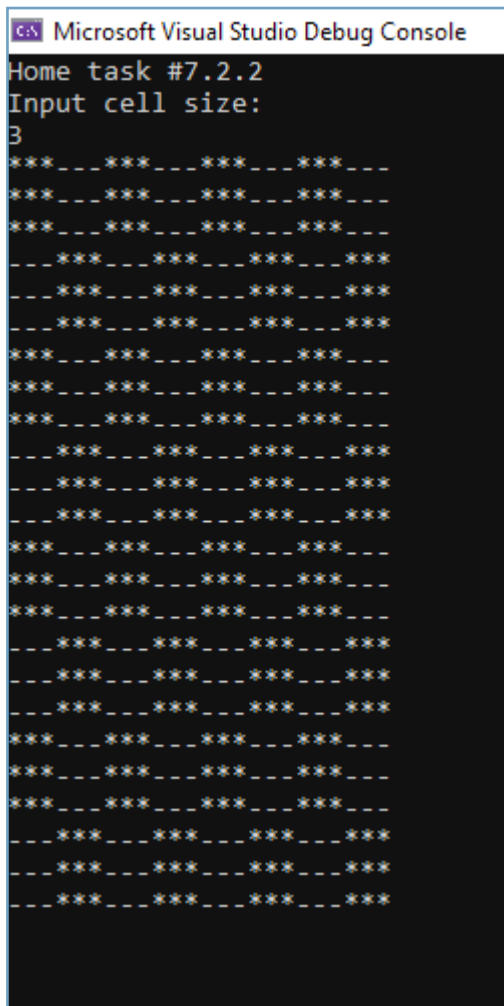
    for (int i = 0; i < 8; i++)
    {
        if (i % 2 == 0)
        {
            for (int m = 0; m < cellSize; m++)
            {
                for (int j = 0; j < 8; j++)
                {
                    if (j % 2 == 0)
                    {
                        for (int k = 0; k < cellSize; k++)
                        {
                            cout << "*";
                        }
                    }
                    else
                    {
                        for (int k = 0; k < cellSize; k++)
                        {
```



```
        cout << "-";
    }
}
cout << "\n";
}
}
else {
    for (int m = 0; m < cellSize; m++)
    {
        for (int j = 0; j < 8; j++)
        {
            if (j % 2 == 0)
            {
                for (int k = 0; k < cellSize; k++)
                {
                    cout << "-";
                }
            }
            else
            {
                for (int k = 0; k < cellSize; k++)
                {
                    cout << "*";
                }
            }
        }
        cout << "\n";
    }
}
return 0;
}
```

## Результаты тестирования программы

- Тест 1 — размер ячейки шахматной доски 3 символа



```
Microsoft Visual Studio Debug Console
Home task #7.2.2
Input cell size:
3
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
***- ***- ***- ***-
```

Рисунок 7

- Тест 2 размер ячейки шахматной доски 5 символов

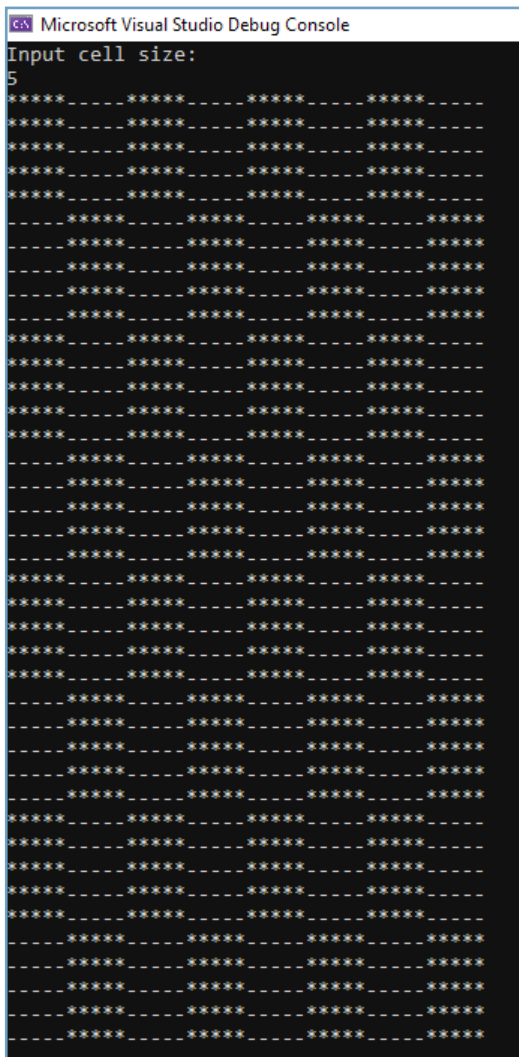


Рисунок 8