

Факультет Информационных систем и технологий
Кафедра Безопасности информационных систем

Дисциплина «Технологии программирования»

ОТЧЕТ

за практическое занятие №2

Тема: Обработка данных в массивах

Выполнил
Студент 1 курса, гр. ИСТ
_____ К.М.

Принял
Доцент кафедры БИС
_____ Медведев В.А.

Тема: обработка данных в массивах.

Цель: закрепление практических навыков разработки программ с использованием операторов циклов при обработке массивов.

Задание: разработать программу, которая определяет, является ли заданный явно массив размерности 9×9 матрицей для квадрата Судоку.

Текст программы:

```
#include <iostream>
using namespace std;
int A[9][9] =
{
    {1,2,3,4,5,6,7,8,9},
    {4,5,6,7,8,9,1,2,3},
    {7,8,9,1,2,3,4,5,6},
    {2,3,4,5,6,7,8,9,1},
    {6,7,8,9,1,2,3,4,5},
    {8,9,1,2,3,4,5,6,7},
    {3,4,5,6,7,8,9,1,2},
    {5,6,8,9,1,2,3,4,7},
    {9,1,2,3,4,5,6,7,8},
};
int p = 0;
int function1()
{
    int i, j = 0;
    int m;
    int y = 0;
    for (A[i][j]; i < 9; i++)
    {
        p = 0;
        m = 0;
        for (A[i][m]; m < 9; m++)
        {
            p += A[i][m];
        }
        if (p == 45)
        {
            y += 1;
        }
        else
        {
            return false;
            break;
        }
    }
}
```

```

    }
    if (y == 9)
    {
        return true;
    }
}
int r = 0;
int function2()
{
    int i, j = 0;
    int m;
    int y = 0;
    for (A[i][j]; j < 9; j++)
    {
        r = 0;
        m = 0;
        for (A[m][j]; m < 9; m++)
        {
            r += A[m][j];
        }
        if (r == 45)
        {
            y += 1;
        }
        else
        {
            return false;
            break;
        }
    }
    if (y == 9)
    {
        return true;
    }
}
int f3(int i[])
{
    int y = 0;
    int u = 0;
    for (A[i][j]; j < w; j++)
    {
        int m = i;
        for (A[m][j]; m < x; m++)
        {

```

```

        u += A[m][j];
    }
}
if (u==45)
{
    y +=1;
}
if (y == 1)
{
    return true;
}
else
{
    return false;
}
}
int main()
{
    if (A[0][0] + A[1][1] != 10)
    {
        int a = function1();
        int b = function2();
        int c = f3(0,0,3,3) + f3(0,3,6,3) + f3(0,6,9,3)
+ f3(3,0,3,6) + f3(3,3,6,6) + f3(3,6,9,6) + f3(6,0,3,9)
+ f3(6,3,6,9) + f3(6,6,9,9);

        if (a + b + c == 11)
        {
            cout << "This is sudoku square";
        }
        else
        {
            cout << "Not sudoku square";
        }
    }
    else
    {
        cout << "Not sudoku square";
    }
}
}

```

Лучше так: взяла массив 9x9, прогнала его по одной функции, взяла массив 3x3, развернула его в одну строку, прогнала по этой же функции.

Результаты работы программы:

Ниже (Рисунок №1 и Рисунок №2) представлены результаты выполненной программы в окне консоли.

```
1 #include <iostream>
2
3 using namespace std;
4 int A[9][9] =
5 {
6     {1,2,3,4,5,6,7,8,9},
7     {4,5,6,7,8,9,1,2,3},
8     {7,8,9,1,2,3,4,5,6},
9     {2,3,4,5,6,7,8,9,1},
10    {5,6,7,8,9,1,2,3,4},
11    {8,9,1,2,3,4,5,6,7},
12    {3,4,5,6,7,8,9,1,2},
13    {6,7,8,9,1,2,3,4,5},
14    {9,1,2,3,4,5,6,7,8}
15 };
16
17
```

Get URL

options compilation execution

This is sudoku square

```
1 #include <iostream>
2
3 using namespace std;
4 int A[9][9] =
5 {
6     {1,2,3,4,5,6,7,8,9},
7     {4,5,6,7,8,9,1,2,3},
8     {7,8,9,1,2,3,4,5,6},
9     {2,3,4,5,6,7,8,9,1},
10    {6,7,8,9,1,2,3,4,5},
11    {8,9,1,2,3,4,5,6,7},
12    {3,4,5,6,7,8,9,1,2},
13    {5,6,8,9,1,2,3,4,7},
14    {9,1,2,3,4,5,6,7,8},
15 };
16 int p = 0;
17
```

Get URL

options compilation execution

Not sudoku square

Рисунок №2

Вывод: в результате выполнения данной практической работы была написана программа, которая определяет, является ли заданный массив матрицей для квадрата Судоку, на языке программирования C++. Для решения поставленной задачи каждые строки, столбцы и квадраты (3x3) были проверены на суммы (=45) с помощью функций **function1**, **function2** и **f3**. Если в какой-то строке/столбце/квадрате сумма не равна 45, значит, числа повторяются, и это не Судоку. Также есть проверка на равенство первых двух элементов строки пятеркам, чтобы избежать ситуаций, когда весь массив Судоку будет заполнен числами 5. С помощью оператора цикла **for** был реализован ряд задач: перебор проверяемых чисел, обращение к определенному элементу массива. Также дополнялся оператор цикла условным оператором **if**, **break** и переменной типа **bool**, с их помощью осуществлялись прекращение работы цикла при нахождении повторений в «Судоку», вывод результата.