

Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»

Навчально-науковий комплекс  
«Інститут прикладного системного аналізу»

Відділення другої вищої та післядипломної освіти

**ЛАБОРАТОРНА РОБОТА №1**  
(варіант 1-27)

з курсу «Програмування»  
на тему «Використання керуючих конструкцій»

Виконав: студентка 3-го курсу  
групи ІС-зпб1  
Шуміліна У.О.

Прийняв: викл. Древаль М.М.

Захищено з оцінкою \_\_\_\_\_  
«    » \_\_\_\_\_ 2016 р.  
\_\_\_\_\_

## 1. Умова завдання

Дано квадратну матрицю порядку  $n$  з цілими елементами. Знайти найменше із значень елементів стовпчика, сума модулів елементів котрого є найбільшою. Якщо таких стовпчиків існує декілька, то взяти перший із них.

## 2. Алгоритм розв'язання завдання

### 1. Створення масиву

- 1.1. Запитуємо у користувача розмір квадратної матриці
- 1.2. Якщо значення розмірності від'ємне, генеруємо масив з випадковими значеннями.
- 1.3. Якщо значення розмірності додатне, питаємо в користувача всі значення поелементно.
- 1.4. Якщо це не може бути масивом, виводимо повідомлення про помилку.

### 2. Для того щоб розв'язати завдання треба вирішити дві під задачі!

- 2.1. Знаходимо стовпчик з максимальною сумою модулів елементів.
- 2.2. Проходячи всі елементи стовпчика з максимальною сумою модулів елементів.

### 3. Виводимо результат.

## 3. Лістинг програми

```
#include "stdafx.h"

#include <iostream>
#include <conio.h>
#include <time.h>

using namespace std;

int n;
int** a;

void iniArray() {
    a = (int**)malloc(n * sizeof(int*));
    for (int i = 0; i < n; i++) {
        a[i] = (int*)malloc(n * sizeof(int));
    }
}

void printArray() {
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cout << "\t" << a[i][j];
        }
        cout << endl;
    }
}

void randArray() {
```

```

        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                //a[i][j] = i + j;
                a[i][j] = rand() % 100;
            }
        }

}

void enterArray() {
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cout << "Vvedit element z " << i << " rjadka ";
            cout << j << " stovpchika:" << endl;
            cin >> a[i][j];
        }
    }
}

void findElement() {
    int max_sum = -1;
    int max_row = -1;
    int tempe;

    for (int i = 0; i < n; i++) {
        tempe = 0;
        for (int j = 0; j < n; j++) {
            tempe += abs(a[j][i]);
        }
        if (tempe > max_sum) {
            tempe = max_sum;
            max_row = i;
        }
    }

    tempe = a[0][max_row];
    int min_col = 0;
    for (int i = 1; i < n; i++) {
        if (tempe > a[i][max_row]) {
            tempe = a[i][max_row];
            min_col = i;
        }
    }

    cout << "Minimalnij element v 'maximalnomu' rjadku a[" << min_col
<< "][";
    cout << max_row << "] = " << tempe;
}

int main() {
    srand(time(NULL));
    cout << "Privet, Glinka!" << endl;
    cout << "Vvedit n:" << endl;
    cin >> n;
    if (n<-1) {
        n = abs(n);
        iniArray();
        randArray();
    } else if (n>1) {

```

```

        iniArray();
        enterArray();
    } else {
        cout << "error!";
        _getch();
        return -1;
    }

    cout << "Ccco-co-congratulations! Os' vin:" << endl;
    printArray();

    findElement();

    _getch();
    return 0;
}

```

## 4. Результати роботи програми

Результати роботи програми подамо у вигляді таблиці контрольних значень (табл. 1).

Таблиця 1. Таблиця контрольних значень

№ тесту	Вхідні дані	Результати
1.	-6	0 1 2 3 4 5 1 2 3 4 5 6 2 3 4 5 6 7 8, $a[0][5] = 5$ 3 4 5 6 7 8 9 4 5 6 7 8 9 5 6 7 8 9 10
2.	-2	66 8 54 32, $a[0][1] = 8$
3.	2 3 5 3, 6 7 8 9 5 1	$a[2][2] = 1$
4.	1	error!
5.	-3	96 57 97 13 34 46, $a[1][2] = 46$ 2 60 60

## 5. Висновки

В ході виконання лабораторної роботи, було розв'язано задачу пошуку елемента, що задовольняє умову завдання, студенткою отримано навички створення алгоритму та написання програми, її тестування та створення протоколу лабораторної роботи. Цей досвід в подальшому допоможе студентці стати топовим програмістом на TopCoder.