Фаховий коледж ракетно-космічного машинобудування

Дніпровського національного університету імені Олеся Гончара

ЗВІТ

з лабораторних робіт

з дисципліни «Технології (Програмування)»

Спеціальність 123 Комп’ютерна інженерія

Група

Виконав І.А.Прокопенко

Перевірив М.М. Гапоненко

2022-2023

ЗМІСТ

Лабораторна робота № 17

ЛАБОРАТОРНА РОБОТА № 17

Мета: Придбати практичні навички роботи з Heap – пам’яттю при складанні та налагодженні програм створення та обробки динамічних матриць з використанням вказівників.

Хід роботи

1 Постановка задачі.

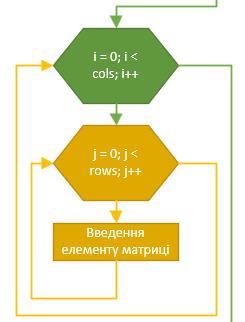
Загальна постановка завдання:

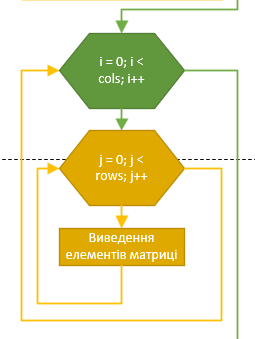
Розмірність матриці (кількість рядків та стовпчиків) користувач вводить з клавіатури. Створити відповідну динамічну матрицю. Обробити її за алгоритмом згідно Вашого варіанта. Програма повинна задовольняти наступним вимогам:

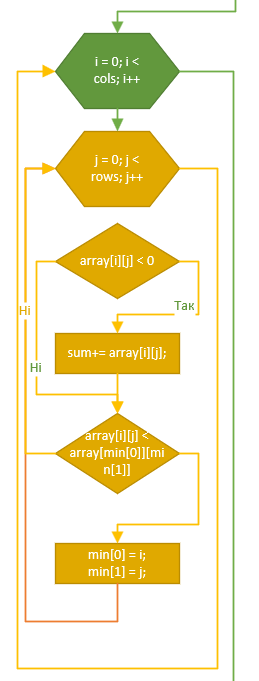
* введення значень елементів матриці проводити в окремому циклі з відповідними коментарями;
* вивести введену матрицю в загальноприйнятому вигляді;
* при обробці матриці враховувати, що шукані елементи можуть бути відсутні. В цьому випадку вивести користувачеві відповідне повідомлення;
* введення та виведення вхідних та вихідних даних повинно містити необхідні для користувача повідомлення.

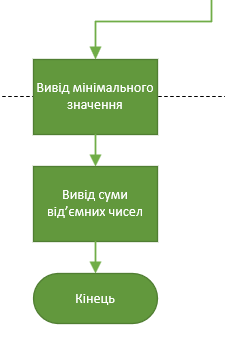
2 Блок-схема алгоритму рішення задачі.









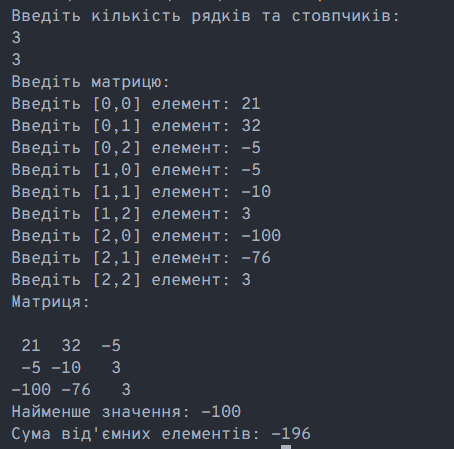


3 Текст програми мовою програмування С++.

|  |
| --- |
| #include <iostream> // для вводу та виводу  #include <stdio.h> // для printf, не обов'язково, але так гарніше виводиться масив  #include <windows.h> // для встановлення кодування  using namespace std;  int main() {      SetConsoleCP(65001); // встановлення кодування      SetConsoleOutputCP(65001); // встановлення кодування      int rows, cols, sum=0; // рядки, стовпчики, та змінна для суми від'ємних елементів      int min[] = {0,0}; // масив для зберігання індексів мінімального числа у матиці      cout << "Введіть кількість рядків та стовпчиків: " << endl;      cin >> rows >> cols;      // Вивільнення місця під масив вказівників      int\*\*array = new int\*[cols];      for (int i = 0; i < cols; i++) {          // створення динамічного масиву елементів у кожному з вказівників          array[i] = new int [rows];      }      // Введення елементів у матрицю      cout << "Введіть матрицю: " << endl;      for (int i = 0; i < cols; i++) {          for (int j = 0; j < rows; j++) {              cout << "Введіть ["<<i<<","<<j<<"] елемент: ";              cin >> array[i][j];          }      }      // Вивід матриці      cout << "Матриця:" << endl;      for (int i = 0; i < cols; i++) {          cout << endl;          for (int j = 0; j < rows; j++) {              printf("%3d ", array[i][j]);              // printf потрібен для виведення чисел з резервацією місця 3              // тобто замість:3 100 5              //          буде:  3 100   5              // це потрібно, аби масив виводився рівно:              //  21  32  -5              //  -5 -10   3          }      }      // Пошук мінімального числа у матриці, а також сума від'ємних елементів      for (int i = 0; i < cols; i++) {          for (int j = 0; j < rows; j++) {              if (array[i][j] < 0)                  sum += array[i][j];              //              if (array[i][j] < array[min[0]][min[1]]) {                  min[0] = i;                  min[1] = j;              }          }      }      cout << endl << "Найменше значення: " << array[min[0]][min[1]] << endl;      cout << "Сума від'ємних елементів: " << sum << endl;      // Вивільнення матриці      // Спочатку ми вивільняжмо масиви      for (int i = 0; i < cols; i++) {          delete[] array[i];      }      // Потім ми вивільняємо масив із вказівниками які раніше вказували на масиви      delete[] array;      // І потім видаляємо адресу вказівника, так як вона нам більше не потрібна      array = NULL;      // Якщо робити це у іншому порядку, то ми не зможемо вивільнити всю зайняту пам'ять, так як втратимо вказівники на елементи      // Якщо не вивільнити матрицю, то у пам'яті можуть залишитись зайняті елементи до наступного перезавантаження      // сучасні системи з цим борються, але раніше це було проблемою        return 0;  } |

4 Копія вікна виконання програми

Рисунок 1 – Зазначення розмірів матриці, введення елементів матриці, виведення суми від’ємних елементів матриці та мінімального елементу матриці



5 Висновок

В результаті виконання лабораторної роботи опрацьовано теоретичний матеріал за темами:

Створення динамичніх матриць та їх обробка.