**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КИЇВСЬКИЙ**

**ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО** »

**Інститут прикладного системного аналізу**

**Кафедра математичних методів системного аналізу**

**Звіт**

**про виконання лабораторної роботи №8**

**з дисципліни «Алгоритмізація та програмування»**

Виконав:

студент I курсу, группи КА-07

Лещинський Богдан Денисович

Прийняв:

к.т.н., доцент Безносик О. Ю.

Київ – 2020

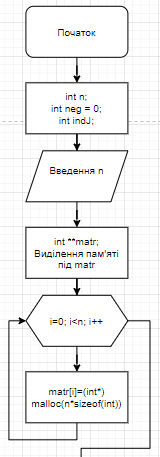
Завдання

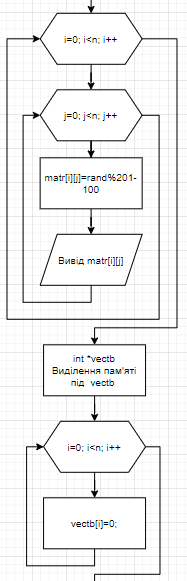
Дана цілочисельна квадратна матриця порядку n. Отримати b1,...,bn, де bi - це сума елементів, попередніх останньому від`ємному елементу i-го рядка (якщо всі елементи рядка невід`ємні, то прийняти bi = -1).

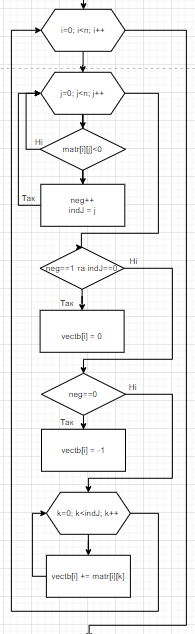
Аналіз умови задачі

Виходячи з умови задачі можна сказати, що алгоритм задачі буде нелінійним, оскільки буде використовуватися динамічний двовимірний масив matr для обробки даних. Для його створення потрібно використати можливості бібліотеки С stdlib.h , а для заповнення випадковими числами – бібліотеки stdlib.h і time.h. Окрім цього, можна сказати, що алгоритм буде циклічним з розгалуженнями, оскільки буде використовуватися вкладені цикли for для перебору значень масиву, і оператори if та else для подальшого заповнення динамічного масиву vectb. Вхідні дані: n - порядок матриці, випадкові елементи matrix . Вихідні дані: елементи масиву vectb. Тип усіх даних – цілі числа int.

Блок-схема алгоритму представлена на рисунку 1:







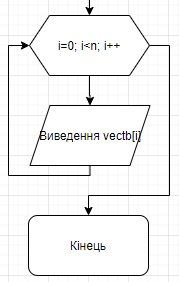


Рис.1

Код программи:

#include<stdio.h>

#include<stdlib.h>

#include<time.h>

int main(){

srand(time(0));

int n;

int neg = 0;

int indJ;

printf("Enter n for a quadratic matrix order: \n");

scanf("%d",&n);

int \*\*matr;

matr = (int\*\*)malloc(n\*sizeof(int\*));

for(int i=0; i<n; i++){

matr[i]=(int\*)malloc(n\*sizeof(int));

}

printf("The quadratic random filled matrix of order %d : \n", n);

for(int i=0; i<n; i++){

for(int j=0; j<n; j++){

\*(matr[i]+j) = rand()%201-100;

printf("%d\t",\*(matr[i]+j));

}

printf("\n");

}

int \*vectb;

vectb = (int\*)malloc(n\*sizeof(int));

for(int i=0; i<n; i++){

vectb[i]=0;

}

for(int i=0; i<n; i++){

for(int j=0; j<n; j++){

if(\*(matr[i]+j)<0){

neg++;

indJ = j;

}

}

if(neg==1 && indJ==0){

vectb[i] = 0;

}

if(neg==0){

vectb[i] = -1;

}

else{

for(int k=0; k<indJ; k++){

vectb[i] += \*(\*(matr+i)+k);

}

}

}

printf("Vector b: \n");

for(int i=0; i<n; i++){

printf("%d\t", vectb[i]);

}

return 0;

}

Результат роботи програми представлентй на рисунку 2:

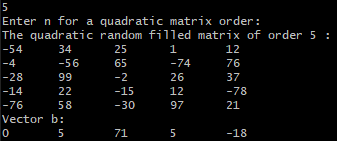


Рис.2

Висновки:

В ході виконання лабораторної роботи №8 було розроблено алгоритм розв’язку задачі знаходження елементів,що є сумою елементів і-того рядка цілочисельної квадратної матриці порядку n, попередніх останньому від’ємному елементу і-того рядка. В програмі використовувався двовимірний динамічний масив цілих чисел int matr для збереження і обробки даних. Для створення динамічного масиву було використано можливості бібліотеки С stdlib.h, а для заповнення випадковими числами – бібліотеки stdlib.h і time.h. Алгоритм є циклічним з розгалуженнями, оскільки використовувався цикл for для перебору значень масиву, і оператори if та else для подальшого заповнення динамічного масиву vectb. Усі дані: масив matr, vectb, змінна n - мають тип даних ціле число int. Для виведення використовується форматоване виведення.