****

**Документация**

**Задача номер : 3**

**Изготвил: Александър Александров Стойчев**

**Фак номер: 361222011**

**Група: 20А**

**Факултет : ФИТ**

**Специалност:ИСИИ**

**Технически университет София**

**Поверена дата на курсовата задача:08.03.2023г.**

**Преподаватели: гл.ас.д-р инж. Кирил Копaранов**

**Доц.д-р инж. Даниела Минковска**

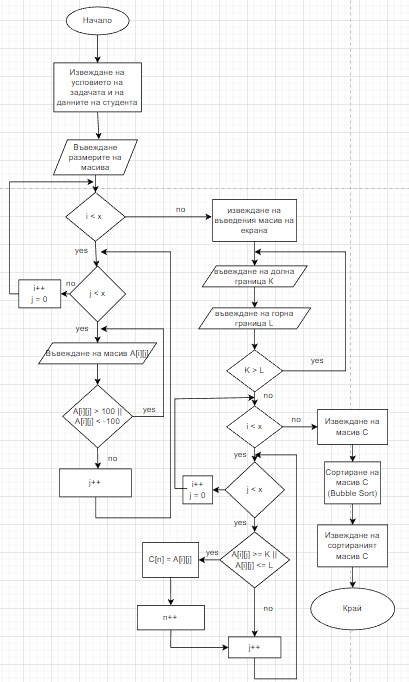
**Задача 3**

**Условие :**

**3.Да се състави програма за обработка на двумерния масив A[N,N], където данните са цели числа в интервала [-100;100]. Програмата да извърши следните действия:**

* **отпечатване на условието на задачата;**
* **отпечатване на имената на автора на програмата;**
* **въвеждане на входните данни;**
* **отпечатване на входните данни;**
* **а) да се образува едномерен масив C, елементите на който са елементите от масива А, попадащи в даден интервал [K - L] ;**
* **б) полученият масив да се сортира по големина;**
* **отпечатване на получените резултати след обработката**

**Блок схема на алгоритма**

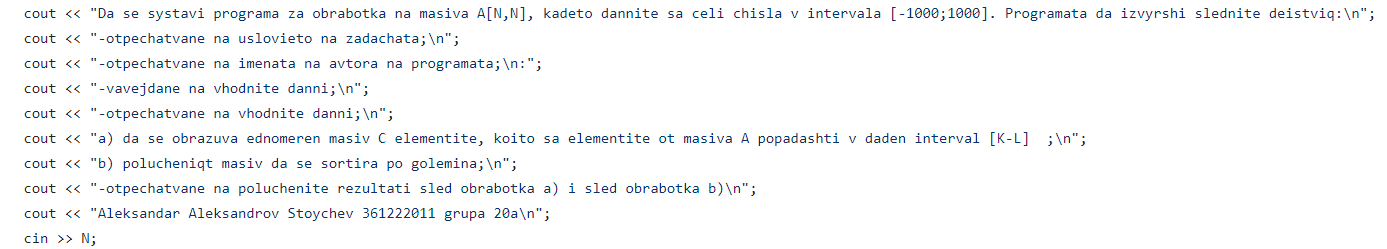


**Описание за функциониране на програмата**

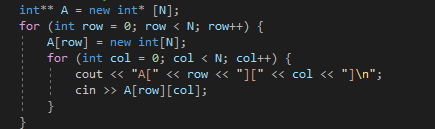
Вход: Въвежда се размерът на масив А, след това се въвеждат стойностите на неговите елементи. След като се въведат всички елементи, се въвежда интервал [K, L].

Изход: Извежда се масив А, масив С, който съдържа всички елементи в интервала [K, L], и сортиран по големина масив С.

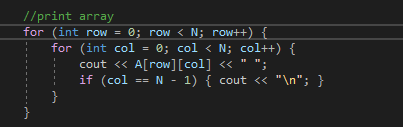
**Части на програмата**



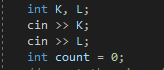
**Извеждане на условието**



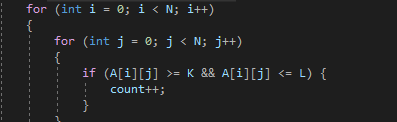
**Декларираме двуизмерен масив А и го попълваме с входните данни**



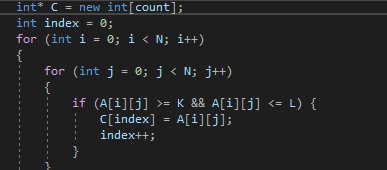
**Отпечатване на масива**



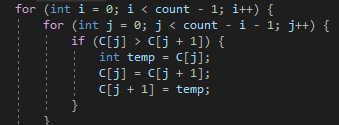
**Декларираме K,L и count**



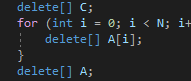
**Преброяваме размера на масива C**



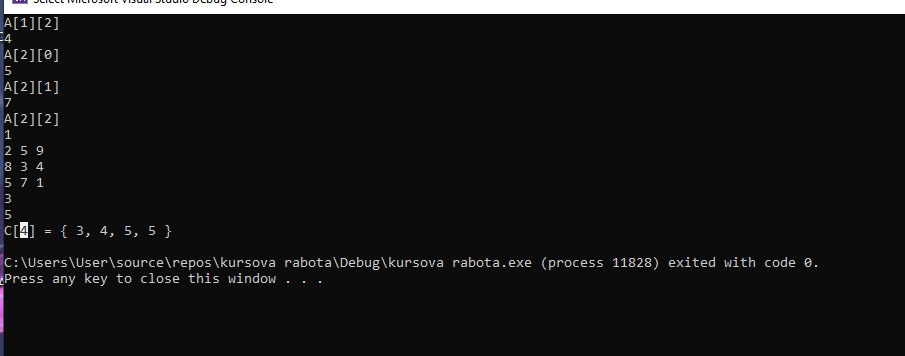
**Добавяме числата към масива**



**Използваме Bubble Sort, за да го подредим по-големина**



**Изтриване на масивите**



**Изходните данни и резултатът от нашата задача**

**Изводи:**

* **Програмата изпълнява задачата поставена в условието.**
* **Програмата работи с реални числа**
* **Програмата работи с отрицателни числа, нула и положителни числа.**
* **Не се позволява въвеждане на стойности извън интервала [-1000, 1000].**

**Сорс кода:**

|  |  |
| --- | --- |
|  | #include <iostream> |
|  |  |
|  | using namespace std; |
|  | int main() |
|  | { |
|  | int N; |
|  |  |
|  | cout << "Da se systavi programa za obrabotka na masiva A[N,N], kadeto dannite sa celi chisla v intervala [-1000;1000]. Programata da izvyrshi slednite deistviq:\n"; |
|  | cout << "-otpechatvane na uslovieto na zadachata;\n"; |
|  | cout << "-otpechatvane na imenata na avtora na programata;\n:"; |
|  | cout << "-vavejdane na vhodnite danni;\n"; |
|  | cout << "-otpechatvane na vhodnite danni;\n"; |
|  | cout << "a) da se obrazuva ednomeren masiv C elementite, koito sa elementite ot masiva A popadashti v daden interval [K-L] ;\n"; |
|  | cout << "b) polucheniqt masiv da se sortira po golemina;\n"; |
|  | cout << "-otpechatvane na poluchenite rezultati sled obrabotka a) i sled obrabotka b)\n"; |
|  | cout << "Aleksandar Aleksandrov Stoychev 361222011 grupa 20a\n"; |
|  | cin >> N; |
|  | // read array |
|  | int\*\* A = new int\* [N]; |
|  | for (int row = 0; row < N; row++) { |
|  | A[row] = new int[N]; |
|  | for (int col = 0; col < N; col++) { |
|  | cout << "A[" << row << "][" << col << "]\n"; |
|  | cin >> A[row][col]; |
|  | } |
|  | } |
|  |  |
|  | //print array |
|  | for (int row = 0; row < N; row++) { |
|  | for (int col = 0; col < N; col++) { |
|  | cout << A[row][col] << " "; |
|  | if (col == N - 1) { cout << "\n"; } |
|  | } |
|  | } |
|  |  |
|  | //a) |
|  | int K, L; |
|  | cin >> K; |
|  | cin >> L; |
|  | int count = 0; |
|  | // count the size of the array |
|  | for (int i = 0; i < N; i++) |
|  | { |
|  | for (int j = 0; j < N; j++) |
|  | { |
|  | if (A[i][j] >= K && A[i][j] <= L) { |
|  | count++; |
|  | } |
|  | } |
|  | } |
|  | // add the numbers to the array |
|  | int\* C = new int[count]; |
|  | int index = 0; |
|  | for (int i = 0; i < N; i++) |
|  | { |
|  | for (int j = 0; j < N; j++) |
|  | { |
|  | if (A[i][j] >= K && A[i][j] <= L) { |
|  | C[index] = A[i][j]; |
|  | index++; |
|  | } |
|  | } |
|  | } |
|  | //á) |
|  | // Buble Sort |
|  | for (int i = 0; i < count - 1; i++) { |
|  | for (int j = 0; j < count - i - 1; j++) { |
|  | if (C[j] > C[j + 1]) { |
|  | int temp = C[j]; |
|  | C[j] = C[j + 1]; |
|  | C[j + 1] = temp; |
|  | } |
|  | } |
|  | } |
|  | // print result |
|  | cout << "C[" << count << "] = { "; |
|  | for (int i = 0; i < count; i++) |
|  | { |
|  | if (i == count - 1) { |
|  | cout << C[i] << " }\n"; |
|  | } |
|  | else { |
|  | cout << C[i] << ", "; |
|  | } |
|  | } |
|  | // delete the arrays |
|  | delete[] C; |
|  | for (int i = 0; i < N; i++) { |
|  | delete[] A[i]; |
|  | } |
|  | delete[] A; |
|  | return 0; |