TALLINNA TEHNIKAÜLIKOOL
INFOTEHNOLOOGIA TEADUSKOND
Arvutisüsteemide instituut

 xxxxxxxxxxxxxxxxxxxxxxxx
IAX0583 Programmeerimine I
**MASSIIVID**

Kodutöö nr.2

Juhendaja: Vladimir Viies,
 dotsent

Tallinn 20XX

**Autorideklaratsioon**Kinnitan, et käesolev töö on minu töö tulemus ja seda ei ole varem esitatud minu ega kellegi teise poolt.

xxxxxxxxxxxx

# Sisukord

[Sisukord 3](#_Toc342252358)

[Ülesande püstitus 4](#_Toc342252359)

[Programmikood 5](#_Toc342252360)

[Programmi selgitus 8](#_Toc342252361)

[Pilt programmist 9](#_Toc342252362)

[Kasutatud kirjandus 10](#_Toc342252363)

# Ülesande püstitus

Variant 949

Klaviatuurilt sisestatakse maatriksi A ridade arv N ja veergude arv M, (1≤ M,N ≤10) ning ridade kaupa reaalarvulised elemendid.

Ekraanile väljastatakse koos oma indeksitega selle veeru suurim element, kus elementide absoluutväärtuste summa on minimaalne.

#

# Programmikood

#include <stdio.h>

#include <math.h>

#define MAXM 10

#define MAXN 10

void Sisestus(int \*N, int \*M, int A[MAXN][MAXM]);

void Sortimine(int N, int M,int A[MAXN][MAXM],int B[M]);

void Sortimine2(int N, int M, int A[MAXN][MAXM],int B[M]);

int main()

{

 int M,N,A[MAXN][MAXM],B[MAXM];

 Sisestus(&M,&N,A);

 Sortimine(N,M,A,B);

 Sortimine2( N,M, A,B);

 return 0;

}

 void Sisestus(int \*N, int \*M, int A[MAXN][MAXM])

 {

 int i=0;

 int j=0;

 printf("Sisesta ridade arv N (1<=N<=10) ");

 scanf("%d", N);

 if(\*N<1 || \*N>10)

 {

 while(\*N<1 || \*N>10){

 printf("Ridade arv peab jaama 1 ja 10 vahele, sisesta ridade arv UUESTI! ");

 scanf("%d", N);

 }

 }

 printf("Sisesta veergude arv M (1<=M<=10) ");

 scanf("%d", M);

 if(\*M<1 || \*M>10)

 {

 while(\*M<1 || \*M>10){

 printf("Veergude arv peab jaama 1 ja 10 vahele, sisesta veergude arv UUESTI! ");

 scanf("%d", M);

 }

 }

 while(i<\*N)

 {

 j=0;

 while(j<\*M)

 {

 printf("Sisesta element A[%d][%d] ",i,j);

 scanf("%d", &A[i][j]);

 j++;

 }

 i++;

 }

 }

void Sortimine(int N, int M,int A[MAXN][MAXM],int B[M])

{

int s=0;

int i, j;

for (j=0;j<M;j++)

{

 for(i=0;i<N;i++)

 {

 s+=fabs(A[i][j]);

 }

 B[j]=s;

 s=0;

}

for(i=0;i<N;i++)

{

 printf("%d veeru summa on %d \n", i+1, B[i]);

}

}

void Sortimine2(int N, int M, int A[MAXN][MAXM],int B[M])

{

 int i,v;

int vi=0;

v=B[0];

for(i=0;i<M;i++)

{ if(B[i]<v)

 {

 v=B[i];

 vi=i;

 }

}

int vs=A[0][vi];

for(i=0;i<N;i++)

{

 if(vs<A[i][vi])

 vs=A[i][vi];

}

 printf("Suurim element on veerus %d arv %d",vi+1,vs);

# Programmi selgitus

Kõigepealt lisatakse teek math.h , et kasutada fabs funktsiooni, millega arvutatakse absoluutväärtust. Defineeritakse ridade ja veergude max väärtused.

Peaprogrammis int main on ainult deklareeritud muutujad, seejärel kutsutakse alamprogramme.

Esimeses alamprogrammis "void Sisestus" sisestatakse ridade arv N ja veergude arv M. Seejärel täidetakse massiiv A [N][M] täisarvuliste elementidega.

Alamprogrammis "void Sortimine" leitakse massiivi A veergude absoluutväärtuste summad ja lisatakse need massiivi B. Seejärel trükitakse välja massiiv B.

"Void Sortimine2" alamprogrammis tuvastatakse, milline massiivi B väärtustest on väikseim ja selle väärtus omistatakse int v -le. Sealjuures omistatakse int vi -le see indeksi väärtus, kus kohapeal oli massiivis B väikseim väärtus. Niisis, kui on teada väikseima elemendi indeksi number massiivis B, on teada ka kõige väiksema absoluutväärtuste summaga veerg massiivis A. Seejärel otsitakse sealt veerust massiivi A kõige väiksem element veerus vi. See omisitatakse int vs -le ja prinditakse lõouks välja.

# Pilt programmist

# progre.JPG

# Kasutatud kirjandus

1. Teodor Luczkowski "Baasteadmised programmeerimiskeelest C++", TTÜ, 2004
2. Allikas http://www.tud.ttu.ee/material/vladimir/ seisuga 2.12.2012