**Programiranje I**

***GRUPA A***

**Integralni ispit - 28.01.2017.**

**Zadatak 1**

Poštujući faze programiranja napisati program za računanje binominalnog koeficijenta ***b*** prema sljedećoj formuli. Učitati prirodne brojeve n i k (n mora biti veći od k). Ako taj uvjet nije ispunjen ponoviti učitavanja. Koristiti posebnu funkciju za računanje binomnog koeficijenta. Naći i ispisati binomni koeficijent koji je jednak sljedećoj formuli:

Prijedlog rješenja:

int fact(int);

double binomni(int, int);

int main()

{

int n,k;

cout<<"Unesi n i k"<<endl;

do

{

cin>>n>>k;

if(n<k||k<0)

cout<<"Ponovite unos "<<endl;

}while (n<k||k<0);

cout<<"Binomni koeficijent je: "<<binomni(n,k)<<endl;

system("pause");

return 0;

}

int fact(int broj)

{

int rez=1;

for(int i=2;i<=broj;i++)

{

rez\*=i;

}

return rez;

}

double binomni(int n, int k)

{

return fact(n)/(fact(k)\*double(fact(n-k)));

}

**Zadatak 2**

Napisati program koji će učitati prirodni broj n<=10, a zatim n cijelih brojeva. Za svaki broj treba ispitati radi li se o broju koji je 'dobar'. Zatim sve dobre brojeve prebaciti u novi niz te ispisati njegove elemente i veličinu. Koristiti zasebne funkcije za unos i provjeru je li broj dobar.

Napomena: Broj je „dobar“ ako je svaka njegova cifra veća od zbira cifara koji se nalaze desno od nje. Na primjer, 9620 je „dobar“ jer je 2>0, 6>2+0, 9>6+2+0. Drugi primjeri: 8421, 95210, 732. 621, 852 itd.

Prijedlog rješenja:

int unos(int[], int);

bool dobar(int);

int main()

{

int niz[10] = {};

int n;

do

{

cout<<"Unesi broj u rangu 1-10: "<<endl;

cin>>n;

if(n<=1||n>=10)

cout<<"Niste unijeli ispravan broj. Ponovite unos"<<endl;

}

while (n<=1||n>=10);

int velicina = unos(niz, n);

cout<<"Velicina niza dobrih brojeva je: "<< velicina<<endl;

cout<<"Njegovi clanovi su: "<<endl;

for(int i = 0; i<velicina; i++)

{

cout<<setw(7)<<niz[i];

}

cout<<endl;

system("pause");

return 0;

}

int unos(int niz[], int n)

{

int brojDobrih=0;

int clan;

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

{

cout<<"Unesi "<<i+1<<". clan niza"<<endl;

cin >>clan;

if (dobar(clan))

{

niz[brojDobrih++]=clan;

}

}

return brojDobrih;

}

bool dobar(int broj)

{

int trenutna, zbirDesnih=broj%10;

broj/=10;

while(broj) //ili while(broj>0)

{

trenutna=broj%10;

if (trenutna<=zbirDesnih)

{

return false;

}

zbirDesnih+=trenutna;

broj/=10;

}

return true;

}

**Zadatak 3**

Napisati glavni program i funkciju *medij* koja se poziva naredbom *medij*(v,n) i koja nalazi **medijan** tj. srednji član u sortiranom nizu v. Ako je n paran, tada je *medijan* aritmetička sredina od srednja dva člana. Ako je n neparan, tada je *medijan* član u sredini (na n/2-om mjestu). U glavnom programu korisnik unosi veličinu niza n (1<=n<=10) te članove niza koje prije proslijeđivanja funkciji sortira, poziva funkciju te ispisom na ekran dobija **medijan**.

Prijedlog rješenja:

float medij(int [], int);

void sortiraj(int [], int);

int main()

{

int v[10] = {};

int n;

do

{

cout<<"Unesi broj u rangu 1-10: "<<endl;

cin>>n;

if(n<=1||n>=10)

cout<<"Niste unijeli ispravan broj. Ponovite unos"<<endl;

}

while (n<=1||n>=10);

cout<<"Unesi elemente niza"<<endl;

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

{

cin>>v[i];

}

sortiraj(v, n);

cout<<"Medijan sortiranog niza je: "<<medij(v,n)<<endl;

system("pause");

return 0;

}

void sortiraj(int niz[], int n)

{

bool promjena = true;

int prolaz = n-1;

while (promjena)

{

promjena = false;

for(int i = 0; i<prolaz; i++)

{

int privremeni;

if (niz[i]>niz[i+1])

{privremeni = niz[i];

niz[i] = niz[i+1];

niz[i+1] = privremeni;

promjena = true;

}

}

prolaz--;

}

}

float medij(int niz[], int n)

{

if(n%2==0)

return (niz[n/2]+niz[n/2-1])/2.0;

else

return float(niz[n/2]);

}

**Zadatak 4**

Napisati program koji će učitati cijeli broj n (1≤n≤10), (ukoliko unesena vrijednost nije unutar intervala, učitavanje treba ponavljati sve dok se ne unese ispravna vrijednost) i elemente matrice dimenzija n x n te ispisati je li matrica centralno simetrična s obzirom na središnji element. Ako jest, program treba ispisati 1, ako nije, ispisati 0, a ako je broj redaka paran broj (pa nema središnjeg elementa), ispisati -1. Koristiti zasebne funkcije za **unos** elemenata i **provjeru simetričnosti**.  
Primjeri centralno-simetričnih matrica:

Prijedlog rješenja:

#include <iostream>

using namespace std;

const int maxstup = 10;

void unos(int mat[][maxstup], int n)

{

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

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

cout << "mat[" << i << "][" << j << "] = ";

cin >> mat[i][j];

}

}

}

int provjera(int mat[][maxstup], int n)

{

int rez = 1;

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

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

if (mat[i][j] != mat[n - 1 - i][n - 1 - j]){

rez = 0;

break;

}

}

}

return rez;

}

int main(){

int mat[maxstup][maxstup];

int n, rez;

do{

cout<< "Unesite veličinu matrice: ";

cin>>n;

} while (n<1 || n>10);

rez = 1;

if (n % 2 == 0){

rez = -1;

cout << rez << endl;

return 0;

}

else{

unos(mat, n);

rez = provjera(mat, n);

}

cout << rez << endl;

system("pause>0");

return 0;

}