

**PMF Kragujevac**  
**Institut za matematiku i informatiku**

seminarski rad

Izračunavanje vrednosti binoma  $(x + y)^n$  uz pomoć Paskalovog trougla  
(Programski jezik C)

Predmet: **Teorijske osnove informatike 3**

student:

Srđan Nikolić 59/08

predmetni asistent:

Tatjana Aleksić

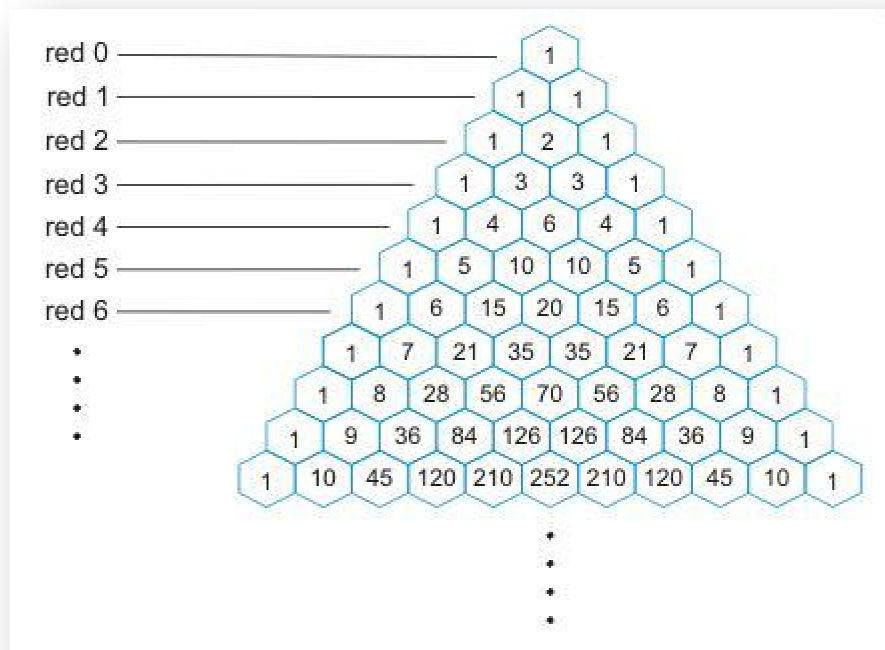
**Zadatak** : Izračunati vrednost binom  $(x + y)^n$ . Binomne koeficijente odrediti formiranjem Paskalovog trougla!

Binomni koeficijenti su koeficijenti u razvoju binoma  $(x + y)^n$  (odatle i naziv):

$$(x + y)^n = \sum_{k=0}^n \binom{n}{k} x^k y^{n-k}$$

## Paskalov trougao

**Paskalov trougao** predstavlja beskonačan niz prirodnih brojeva, koji je u obliku piramidalne šeme. Svaki broj u jednom redu predstavlja zbir brojeva koji su iznad njega. Krajnji brojevi šeme su uvek jedinice. Ovi brojevi posmatrani po vrstama ponašaju se kao **binomni koeficijenti** (ovo je ključna stvar u resavanju problema).



\*Naziv je dobio po matematičaru **Blezu Paskalu**.

Dakle, uz pomoć Paskalovog trougla lako mozemo odrediti binomne koeficijente.

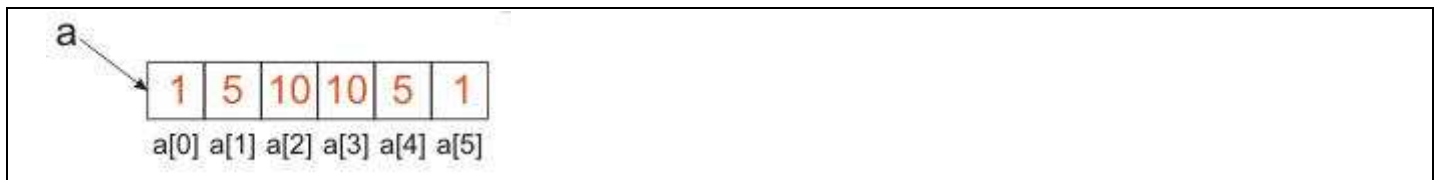
Ako pogledamo 5. red lako mozemo doći do razvijenog oblika binoma:

$$(x + y)^5 = 1x^5 + 5x^4y + 10x^3y^2 + 10x^2y^3 + 5xy^4 + 1y^5.$$

### Kako problem rešiti u računaru?

Za početak, napisaćemo funkciju koja generiše binomne koeficijente, tačnije, napisaćemo funkciju koja generiše niz binarnih koeficijenata. Dakle, ukoliko imamo binom stepena **n**, funkcija će vratiti niz brojeva koji će predstavljati binomne koeficijente **n+1** vrste Paskalovog trougla.

Za **n=5**, nama je cilj da formiramo sledeći niz:



Generisanje `n`-te vrste Paskalovog trougla ćemo vršiti u jednom nizu čija je dužina "`n+1`"(to se lako može uočiti sa slike Paskalovog trougla), i koristiceemo osobinu koju imaju svi brojevi u trouglu, a to je da svaki broj u jednom redu predstavlja zbir brojeva koji su iznad njega.

```

long* pascalov_trougao(int stepenBinoma)
{
    int i,j;
    long *a, pom1, pom2;

    a = (long*)calloc(stepenBinoma+1, sizeof(long));

    //formiranje Pascalovog trougla
    for(i=0; i<=stepenBinoma; i++)
    {
        pom1 = 0;
        pom2 = 1;
        for(j=0; j<=i; j++)
        {
            a[j] = pom1 + pom2;
            pom1 = pom2;
            pom2 = a[j+1];
        }
    }
    return a;
}

```

**Primer.** Šta se dešava u memoriji ukoliko funkciji pošaljemo broj **5**?

```

a = (long*)calloc(stepenBinoma + 1, sizeof(long));

```



**1. iteracija ( i = 0, pom1=0, pom2=1 )**

```

j = 0;

```

```

a[0] = 0 + 1 = 1;

```



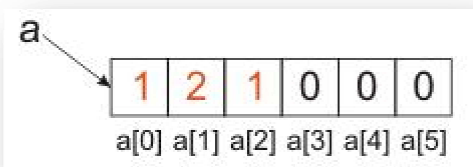
2. iteracija (  $i = 0$ ,  $pom1=0$ ,  $pom2=1$  )

<code>j = 0;</code> <code>a[0] = 0 + 1 = 1;</code> <code>pom1 = 1;</code> <code>pom2 = 0;</code>	<code>j = 1;</code> <code>a[1] = 1 + 0 = 1;</code>
---	---



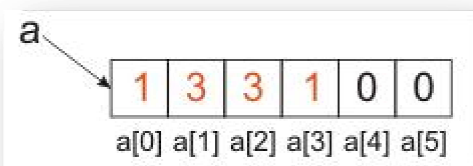
3. iteracija (  $i = 0$ ,  $pom1=0$ ,  $pom2=1$  )

<code>j = 0;</code> <code>a[0] = 0 + 1 = 1;</code> <code>pom1 = 1;</code> <code>pom2 = 1;</code>	<code>j = 1;</code> <code>a[1] = 1 + 1 = 2;</code> <code>pom1 = 1;</code> <code>pom2 = 0;</code>	<code>j = 2;</code> <code>a[2] = 1 + 0 = 1;</code>
---	---	---



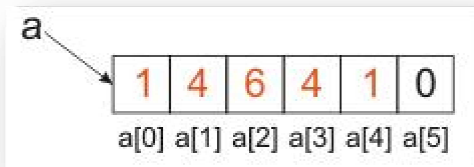
4. iteracija (  $i = 0$ ,  $pom1=0$ ,  $pom2=1$  )

<code>j = 0;</code> <code>a[0] = 0 + 1 = 1;</code> <code>pom1 = 1;</code> <code>pom2 = 2;</code>	<code>j = 1;</code> <code>a[1] = 1 + 2 = 3;</code> <code>pom1 = 2;</code> <code>pom2 = 1;</code>	<code>j = 2;</code> <code>a[2] = 2 + 1 = 3;</code> <code>pom1 = 1;</code> <code>pom2 = 0;</code>	<code>j = 3;</code> <code>a[3] = 1 + 0 = 1;</code>
---	---	---	---



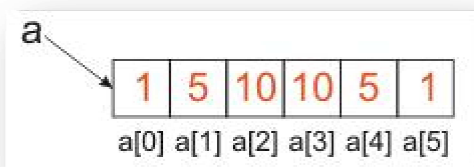
### 5. iteracija ( i = 0, pom1=0, pom2=1 )

j = 0; a[0] = 0 + 1 = 1; pom1 = 1; pom2 = 3;	j = 1; a[1] = 1 + 3 = 4; pom1 = 3; pom2 = 3;	j = 2; a[2] = 3 + 3 = 6; pom1 = 3; pom2 = 1;	j = 3; a[3] = 3 + 1 = 4; pom1 = 1; pom2 = 2;	j = 4; a[4] = 1 + 0 = 1;
---	---	---	---	-----------------------------



### 6. iteracija ( i = 0, pom1=0, pom2=1 )

j = 0; a[0] = 0 + 1 = 1; pom1 = 1; pom2 = 4;	j = 1; a[1] = 1 + 4 = 5; pom1 = 4; pom2 = 6;	j = 2; a[2] = 4 + 6 = 10; pom1 = 6; pom2 = 4;	j = 3; a[3] = 6 + 4 = 10; pom1 = 4; pom2 = 1;	j = 4; a[4] = 4 + 1 = 5; pom1 = 1; pom2 = 0;	j = 5; a[5] = 1 + 0 = 1;
---	---	--	--	---	-----------------------------



Zatim je potrebno napisati funkciju koja računa vrednost binoma. Toj funkciji je potrebno poslati niz binomnih koeficijenata i vrednosti parametara a i b.

```
long vrednost_binoma(long nizKoeficijenata[], int n, long a, long b)
{
    long S = 0;
    int i, j;

    for(i=0, j=n; i<=n; i++, j--)
    {
        S += nizKoeficijenata[i]*stepen(a, j)*stepen(b, i);
    }

    return S;
}
```

**Primer.**  $(x + y)^5 = 1x^5 + 5x^4y + 10x^3y^2 + 10x^2y^3 + 5xy^4 + 1y^5$ .

Funkcija **vrednost\_binoma** će u svakoj iteraciji računati vrednosti sabiraka u razvijenom obliku binoma, redom, i tu vrednost će dodavati na sumu.

kod

```
#include <stdio.h>
#include <stdlib.h>

/*Funkcija za racunanje vrednosti n-tog stepena broja
Ulazni parametri
  -broj ciji se stepen trazi, stepen koji se trazi
Izlazni parametar
  -vrednost n-tog stepena broja*/
long stepen(int a, int n)
{
    long pom=1;
    int i;

    for(i=0; i<n; i++)
        pom *= a;

    return pom;
}

/*Funkcija koja formira niz koeficijenata binoma metodom Pascalovog trougla
Ulazni parametri
  -Stepen binoma (int)
Izlazni parametri
  -Adresa prvog elementa niza(pokazivac na long tip podataka)*/
long* pascalov_trougao(int stepenBinoma)
{
    int i,j;
    long *a, pom1, pom2;

    a = (long*)calloc(stepenBinoma+1, sizeof(long));

    //formiranje Pascalovog trougla
    for(i=0; i<=stepenBinoma; i++)
    {
        pom1 = 0;
        pom2 = 1;
        for(j=0; j<=i; j++)
        {
            a[j] = pom1 + pom2;
            pom1 = pom2;
            pom2 = a[j+1];
        }
    }
    return a;
}
```

```

/*funkcija koja racuna vrednost binoma
Ulazni parametri
  -niz vrednosti koeficijenata binoma u razvijenom obliku(long),
  a(long), b(long), stepen binoma(int)
Izlazni parametar
  -vrednost binoma(float)*/
long vrednost_binoma(long nizKoeficijenata[], int n, long a, long b)
{
    long S = 0;
    int i, j;

    for(i=0, j=n; i<=n; i++, j--)
    {
        S += nizKoeficijenata[i]*stepen(a, j)*stepen(b, i);
    }

    return S;
}

main()
{
    long *nizKoeficijenata, rezultat, a, b;
    int stepenBinoma;

    printf("\n (a + b)^n\n");

    printf("\n Unesite vrednost za \"a\": ");
    scanf("%ld",&a);

    printf("\n Unesite vrednost za \"b\": ");
    scanf("%ld",&b);

    printf("\n Unesite vrednost za \"n\": ");
    scanf("%d",&stepenBinoma);

    nizKoeficijenata = pascalov_trougao(stepenBinoma);

    rezultat = vrednost_binoma(nizKoeficijenata, stepenBinoma, a, b);

    printf("\n Vrednost binoma je: %ld",rezultat);

    printf("\n\n");

    return 0;
}

```

```

(a + b)^n
Unesite vrednost za "a": 5
Unesite vrednost za "b": 6
Unesite vrednost za "n": 7
Vrednost binoma je: 19487171

```