

# Generisanje permutacija bez ponavljanja

## 1. Definicija

Permutacija bez ponavljanja skupa  $A_n$  je svaka uredjena  $n$ -torka razlicitih elemenata tog skupa.

Broj permutacija skupa od  $n$  elemenata iznosi  $n!$ .

## 2. Algoritam

$A_4 = \{1, 2, 3, 4\}$

1234	2134	3124	4123
1243	2143	3142	4132
1324	2314	3214	4213
1342	2341	3241	4231
1423	2413	3412	4312
1432	2431	3421	4321

Na ovom primeru se moze uociti da se prvi clan skupa  $A_n$  menja na svakoj sestojoj permutaciji, drugi clan na svakoj drugoj dok treci i cetvrti menjaju na svakoj sledecoj. Sto znaci poslednji clan se menja na svakih  $0!$  permutacija, predposlednji na  $1!$ , sledeci na  $2!$  itd. Dakle, ukoliko je redni broj permutacije deljiv sa  $k!$  to znaci da je doslo vreme da se promeni  $(n-k)$ -ti clan skupa. Ostalo je samo jos da utvrdimo na koji nacin cemo vrsiti promenu. Promenu elementa vrsimo tako sto medju elementima "desno" od njega trazimo najblizi sledeci element. Pod sledecim elementom se misli da je dvojka sledeci jedinici, cetvorka trojci ali i jedinica cetvorci i tako ukrug (na primeru  $A_4$ ).

Uzmimo ponovo skup  $A_4$ . Neka je trenutna permutacija 2341. Njen redni broj je 9 (broji se od nule). Permutaciju 10 dobijamo tako sto :

Ispitamo da li  $(4-1)!$  deli 10. Ne.

Ispitamo da li  $(4-2)!$  deli 10. Da. Dakle menjamo drugi clan (trojku) sa najblizim sledecim desno od njega (sa cetvorkom) i dobijamo 2431.

Ispitamo da li  $(4-3)!$  deli 10. Da. Dakle menjamo treci clan (trojku) sa najblizim sledecim desno od njega (sa jedinicom) i dobijamo 2413.

Kraj.

Odstupanja od ovog pravila se javljaju kod  $(n-1)$ . clana na svakih  $3!$ , kod  $(n-2)$ . clana na svakih  $4!$  itd. U tim slucajevima ih ne treba menjati sa  $n$ -tim clanom.

### 3. Program

U glavnom delu programa se obavlja provera da li je vreme za promenu clana i ako jeste poziva se funkcija "MENJAZA" koja vrši promenu. Pored funkcije "MENJAZA" postoji i funkcija "FAKT" koja samo racuna faktorijel.

```
#include<stdio.h>
int p,n,a[50];

long FAKT(int f){
    int i,q=1;
    for (i=1;i<=f;i++) q=q*i; return q;
}
void MENJAZA(int clan){
    int b=0,i,j,poz=clan,pom,q;
    q=n;
    for (i=1;i<=4;i++)    if ((p % FAKT(n-clan+i))==0) q=n-i;
    //linija iznad se stara o odstupanjima

    pom=(a[clan] % n)+1;
    while (1){ if (clan+1>=q) b=1;
    for (j=clan+1;j<q;j++)
        if (a[j]==pom){poz=j;b=1;break;}
    if (b) break; else pom=(pom % n)+1;
    }
    pom=a[clan]; a[clan]=a[poz]; a[poz]=pom;
}
main() {
int f,k,m,i,j;
long faktI,ukupno;
scanf("%d",&n);
ukupno=FAKT(n);

for (i=0;i<n;i++) a[i]=i+1;
for (k=0;k<n;k++) printf("%d  ",a[k]);

for (p=1;p<ukupno;p++){
    printf("\n");
    faktI=ukupno;
    for (j=0;j<=n-2;j++){
        faktI=faktI/(n-j);
        if ((p % faktI)==0) MENJAZA(j);
    }
for (k=0;k<n;k++) printf("%d  ",a[k]);
}}
```