Kako se mogu klasifikovati neuređeni izbori elemenata?

Neuređeni izbori elemenata su važan koncept u diskretnoj matematici, često korišćen u kombinatornim problemima gdje redoslijed izbora nije važan. Ova klasifikacija uključuje izbore gdje redoslijed nije bitan, a može se dodatno dijeliti **na neuređene izbore bez ponavljanja** i **neuređene izbore sa ponavljanjem**. Razumijevanje ovih kategorija omogućava efikasno rješavanje problema kao što su formiranje timova ili raspodjela resursa.

1. Neuređeni izbor bez ponavljanja (Kombinacije bez ponavljanja)

Definicija: Kombinacije bez ponavljanja su izbori elemenata iz skupa gdje nijedan element ne može biti izabran više puta, a redoslijed izbora nije bitan. Ovaj koncept se često koristi u problemima koji uključuju odabir podskupova iz većeg skupa, gdje nas ne zanima redoslijed elemenata, već samo koje smo elemente izabrali.

Formula: Broj mogućih kombinacija kada biramo k elemenata iz skupa sa n elemenata računa se pomoću binomnog koeficijenta:

$$C(n,k) = \frac{n!}{k!(n-k)!}$$

Ova formula nam govori koliko različitih načina možemo izabrati k elemenata iz skupa sa n elemenata.

Primjer: Ako imamo skup {A, B, C, D} i želimo da izaberemo 2 elementa, broj različitih kombinacija je:

$$C(4,2) = \frac{4!}{2!(4-2)!} = \frac{4 \cdot 3}{2 \cdot 1} = 6$$

Moguće kombinacije su: {A, B}, {A, C}, {A, D}, {B, C}, {B, D}, {C, D}.

2. Neuređeni izbor sa ponavljanjem (Kombinacije sa ponavljanjem)

Definicija: Kombinacije sa ponavljanjem omogućavaju da biramo elemente iz skupa sa ponavljanjem. Ovaj koncept se koristi kada se dozvoljava da se isti element izabere više puta.

Formula: Broj kombinacija sa ponavljanjem može se izračunati pomoću sledeće formule:

$$C(n+k-1,k) = \frac{(n+k-1)!}{k!(n-1)!}$$

Gdje je n broj elemenata u skupu, a k broj izbora.

Primjer: Ako imamo 3 vrste sladoleda (n = 3) i želimo da izaberemo 5 kugli (k = 5), računamo broj mogućih kombinacija sa ponavljanjem:

$$C(3+5-1,5) = C(7,5) = \frac{7!}{5!(7-5)!} = 21$$

Moguće kombinacije uključuju izbore kao što su {A, A, B, B, C} ili {A, A, A, B, C}.

3. Permutacije sa ponavljanjem

Definicija: Iako se permutacije obično odnose na uređen izbor elemenata, u kontekstu permutacija sa ponavljanjem, dozvoljeno je da se neki elementi iz multiskupa ponove. Cilj je da se izračuna koliko različitih rasporeda elemenata možemo dobiti kada se neki elementi ponavljaju. Ključna karakteristika ovde je da redoslijed elemenata jeste bitan, ali se neki elementi mogu pojaviti više puta, što znači da imamo ograničenu fleksibilnost u raspoređivanju tih elemenata.

Primjer: Ako imamo skup {A, A, B}, postoji manje jedinstvenih načina da se ovi elementi permutuju u odnosu na skup gdje su svi elementi različiti. Broj permutacija sa ponavljanjem za multiskup sa n elemenata gde se neki elementi ponavljaju može se izračunati pomoću sljedeće formule:

$$P = \frac{n!}{k_1!k_2!\ldots k_r!}$$

Gdje su $k_1,k_2,\ldots,k_r$ brojevi ponavljanja različitih elemenata.

Primjer: Ako imamo skup {A, A, B}, broj permutacija sa ponavljanjem je:

$$P = \frac{3!}{2!} = \frac{6}{2} = 3$$

Tri različite permutacije su: {A, A, B}, {A, B, A}, {B, A, A}.

Permutacije sa ponavljanjem spadaju u ovu grupu jer se ponašaju slično neuređenim izborima, jer određeni izbori postaju manje značajni zbog ponavljanja istih elemenata.

Kombinacije sa i bez ponavljanja

Kombinacije su neuređeni izbori elemenata iz skupa, ali mogu biti sa ili bez ponavljanja:

1. Kombinacije bez ponavljanja:

a. Kombinacije bez ponavljanja su podskupovi skupa u kojima svaki element može biti izabran samo jednom.
b. Formula za broj kombinacija bez ponavljanja je:

$$C(n,m) = \binom{n}{m} = \frac{n!}{m!(n-m)!}$$

Ovde, n je broj elemenata u skupu, a m je broj elemenata koji biramo.

Primer: Ako imamo skup $S=\{1,2,3,4\}$ i želimo da izaberemo 2 elementa, moguće kombinacije su $\{1,2\},\{1,3\},\{1,4\},\{2,3\},\{2,4\},\{3,4\}$. Broj ovih kombinacija je:

$$C(4,2) = \frac{4!}{2!(4-2)!} = 6$$

Kombinacije sa ponavljanjem:

- Kod kombinacija sa ponavljanjem, svaki element iz skupa može biti izabran više puta.
- Formula za broj kombinacija sa ponavljanjem je:

$$C(n+m-1,m) = \frac{(n+m-1)!}{m!(n-1)!}$$

Primer: Ako imamo skup $S=\{a,b,c\}$ i biramo 2 elementa sa ponavljanjem, kombinacije su $\{a,a\},\{a,b\},\{a,c\},\{b,b\},\{b,c\},\{c,c\}$. Broj ovih kombinacija je:

$$C(3+2-1,2) = \frac{4!}{2!2!} = 6$$

Algoritam za generisanje kombinacija bez ponavljanja:

Kombinacije bez ponavljanja mogu se generisati rekurzivno.

```python
def generate_combinations(skup, m):
    if m == 0:
        return [[]]
    if len(skup) < m:
        return []

    prvi = skup[0]
    ostatak_skupa = skup[1:]

    kombinacije = generate_combinations(ostatak_skupa, m - 1)
    for k in kombinacije:
        k.insert(0, prvi)

    kombinacije_rekurzivni_poziv = generate_combinations(ostatak_skupa, m)

    return kombinacije + kombinacije_rekurzivni_poziv

```

Algoritam za generisanje kombinacija sa ponavljanjem:

```python
def generate_combinations_with_repetition(skup, m):
    if m == 0:
        return [[]]

    kombinacije = []
    for i in range(len(skup)):
        trenutni = skup[i]
        for k in generate_combinations_with_repetition(skup[i:], m - 1):
            kombinacije.append([trenutni] + k)

    return kombinacije
```

Kombinacije sa i bez ponavljanja imaju značajne primene u raznim oblastima računarstva i programiranja:

1. **Optimizacija**: Kombinacije se koriste u rešavanju problema optimizacije kao što su "knapsack problem" i "subsetsum". U ovim slučajevima, potrebno je generisati sve moguće podskupove skupa opcija da bi se pronašlo optimalno rešenje. Na primer, u "knapsack" problemu biraju se podskupovi predmeta tako da njihova ukupna težina ne prelazi kapacitet ranca, dok vrednost izabranih predmeta bude maksimalna.
2. **Testiranje softvera**: Kombinatorno testiranje koristi m-kombinacije za kreiranje seta testova koji pokrivaju sve moguće kombinacije ulaznih podataka. Ovo je posebno korisno u

automatizovanom testiranju, gde je potrebno generisati sve kombinacije ulaza kako bi se osiguralo da softver funkcioniše ispravno u svim mogućim scenarijima.

3. **Kriptografija**: Kombinacije se često koriste za generisanje mogućih ključeva za šifrovanje i dešifrovanje podataka. Kriptografi analiziraju otpornost algoritama na napade tako što procenjuju sve moguće kombinacije ključnih vrednosti koje bi potencijalni napadač mogao koristiti u pokušaju kompromitovanja sistema.
4. **Analiza podataka**: U velikim skupovima podataka, kombinacije omogućavaju istraživanje podskupova i odnosa među podacima. Na primer, analitičari mogu generisati različite kombinacije karakteristika (feature subsets) kako bi pronašli optimalnu grupu za treniranje modela mašinskog učenja.

m-kombinacije multiskupa elemenata

m-kombinacije multiskupa (ili kombinacije sa ponavljanjem) odnose se na načine na koje možemo odabrati **m** elemata iz multiskupa, gdje je moguće ponavljanje elemenata.

- Multiskup je kolekcija elemata u kojoj su dozvoljeni ponovljeni elementi (za razliku od skupa gdje svaki element može biti prisutan samo jednom)

Broj m-kombinacija multiskupa od n elemenata može se izračunati formulom:

$$\bar{C}_n^m = \binom{n+m-1}{m}$$

Dokaz:

Neka je:

$\binom{M}{m} = \{M_1 : |M_1| = m \wedge M^1 \subseteq M\}$

$A = \{(a^1, \ldots, a_m{}^{+1-1}) \in \{0,1\}^{m+l-1} : \{\{a^1, \ldots, a_m{}^{+1-1}\}\} = [0,1]_l{}^{-1}, {}_m\}$.

Znači, $\binom{M}{m}$ je skup svih m-točlanih podmultiskupova od M, a A je skup svih uređenih torki dužine m+l-1 koje imaju tačno m komponenti jednakih 1 i preostalih l-1 komponenti jednakih 0.

Definišemo preslikanje

$$\varphi_{(b_1,\ldots,b_l)} : \binom{M}{m} \to A$$

na sljedeći način

$$\varphi_{(b_1,\ldots,b_l)}(M_1) = (c_1, \ldots, c_{m+l-1}),$$

gdje je $[M_1 = b_1, \ldots, b_l]|_{m1}, \ldots, {}_{ml}$ i za svako $i \in \{1, \ldots, l-1\}$

$$c_{(i-1)_+(m_1+1)} = 0 \quad c_{(i-1)+j} = 1, 0 < j \leq m_i$$

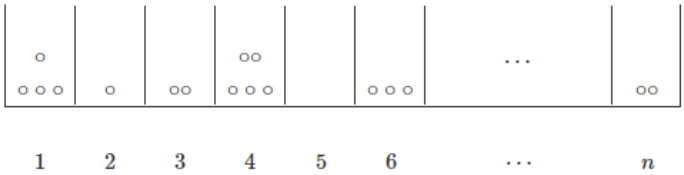
Kako je $\varphi(b_1, \ldots, b_l)$ bijektivno preslikavanje,

$$\bar{C}(l;m) = \left|\binom{M}{m}\right| = |A|$$

Broj načina da od m+l-1 mjesta izaberemo m za 1 i preostalih l-1 za 0 jednak je:

$$\bar{P}(m, l-1) = \frac{(m+l-1)!}{m!\,(l-1)!}$$

II način:
Neka iz skupa od **n** elemenata vršimo odabir **m** puta, pri čemu isti element možemo izabrati i više puta. Zamislimo da imamo **n** praznih kutija (znači, onoliko kutija koliko ima elemenata skupa iz kojeg vršimo odabir), poređanih u niz i priljubljenih jedna uz drugu, tako da su granice između njih pregradni zidovi. Zamislimo, zatim, i da imamo **m** kuglilca, znači, onoliko kuglica koliko puta vršimo odabir. I, na kraju, zamislimo da svaki put kad odaberemo i-ti element tog skupa, ubacujemo jednu kuglicu u i-tu kutiju. Znači, ako smo izabrali 3. element, ubacujemo kuglicu u 3. kutiju. Ako smo zatim izabrali 5. element, ubacujemo sledeću kuglicu u 5. kutiju. Ako smo nakon toga opet izabrali 3. element, ubacujemo još jednu kuglicu u 3. kutiju, tako da će u 3. kutiji sada biti dve kuglice itd... Koliko kuglica imamo u nekoj kutiji, to znači da smo toliko puta izvukli odgovarajući element. Nakon **m** izvlačenja, u svim kutijama će ukupno biti **m** kuglica, a njihov broj u svakoj od kutija pokazivaće koliko smo puta izvukli odgovarajući element.



Sada se traženje formule za m-kombinacija multiskupa elemenata svodi na traženje formule za broj načina na koje možemo **m** kuglica rasporediti u **n** kutija. Na gornjoj slici vidimo jedan od takvih načina, a potrebno je, dakle, odgovoriti na pitanje, koliko ukupno ima različitih načina da se tih **m** kuglica rasporede u tih **n** kutija.
Sada te kuglice u kutijama predstavimo na sledeći način. Napravimo niz od simbola ∘ i |, pri čemu simbol ∘ predstavlja kuglicu, a simbol | predstavlja pregradni zid između dve kutije. Pri tome, levi zid prve i desni zid poslednje kutije ignorišemo. Gornja situacija s kutijama i kuglicama, predstavljena na ovaj način, izgledala bi, dakle, ovako:

∘∘∘∘|∘|∘∘|∘∘∘∘∘||∘∘∘|⋯|∘∘

Pošto kutija ukupno imamo **n**, znači da ćemo pregradnih zidova ukupno imati **n−1**. Kuglica imamo **m**. Znači, kuglica i pregradnih zidova (tj. simbola ∘ i simbola |) ukupno imamo n+m−1. Traženi broj m-kombinacija multiskupa ovime smo sveli na broj načina na koje možemo **m** kuglica rasporediti na **n+m−1** pozicija. Broj načina da se **m** kuglica rasporedi u **n** kutija je broj kombinacija bez ponavljanja od **n + m - 1** pozicija, od kojih biramo **m** pozicija za kuglice:. Prema tome, dobili smo formulu za broj m-kombinacija multiskupa.

Primjer:

Kombinacija s ponavljenjem od 3 elementa {A,B,C} 4. klase (što znači da biramo 4 elementa s ponavljanjem, ne vodeći pri tome računa o njihovom poretku) izgledao bi ovako:

AAAA AABB ABBB ACCC BBCC

AAAB AABC ABBC BBBB BCCC

AAAC AACC ABCC BBBC CCCC

Odnosno računski:

$$n = 3, k = 4 \rightarrow \bar{C}_3^4 = \binom{3+4-1}{4} = \binom{6}{4} = \frac{6!}{(6-4)! * 4!} = \frac{6 * 5 * 4!}{2! * 4!} = \frac{6 * 5}{2} = 15$$

Da li se neuređeni izbori elemenata mogu opisati tj. prebrojati pomoću uređenih izbora elemenata

Ako imamo skup sa n elemenata i želimo da izaberemo k elemenata, broj **uređenih izbora** (permutacija) k elemenata iz skupa n je:

$$P(n,k)=n!/(n-k)!$$

Broj **neuređenih izbora** (kombinacija) k elemenata iz skupa n je:

$$C(n,k)=n!k/!(n-k)!$$

Dakle, broj neuređenih izbora se razlikuje od broja uređenih izbora za faktor k!, što je broj različitih načina na koje možemo rasporediti k elemenata (jer uređenje više nije važno kod kombinacija). U suštini, broj kombinacija (neuređenih izbora) dobijamo tako što podelimo broj permutacija (uređenih izbora) sa brojem načina na koji možemo urediti k elemenata:

$$C(n,k)=P(n,k)/k!$$

Dokaz:

Pretpostavimo da možemo da dobijemo neuređene izbore iz uređenih. Ako imamo skup S sa n elemenata, broj uređenih izbora k elemenata iz S iznosi P(n,k). Kako bi eliminisali duplikate nastale zbog različitih uređenja istih elemenata, delimo broj permutacija sa brojem različitih uređenja, k!, čime dobijamo tačan broj neuređenih izbora:

$$C(n,k)=P(n,k)/k!$$

Ovo pokazuje da je broj **neuređenih izbora** ekvivalentan broju **uređenih izbora** podeljenom sa brojem načina na koji se elementi mogu urediti. Dakle, možemo prebrojati neuređene izbore elemenata pomoću uređenih izbora elemenata.

Zaključak:

Neuređeni izbori (kombinacije) mogu se prebrojati pomoću uređenih izbora (permutacija), jer se broj neuređenih izbora dobija deljenjem broja uređenih izbora sa faktorijalom broja izabranih elemenata k!.

Iz skupa {a,b,c,d} izaberite 2 elementa. Koliko ima:

1. Uređenih izbora (permutacija)?
2. Neuređenih izbora (kombinacija)?

Rešenje:

1. Broj uređenih izbora (permutacija):

Imamo 4 elementa (n=4) i biramo 2 elementa (k=2).

Broj permutacija (uređenih izbora) P(4,2) je dat formulom:

$P(4,2) = 4!/(4-2)! = 4\times3\times2\times1/2\times1 = 4\times3 = 12$

Dakle, postoji 12 uređenih izbora. Oni su:

(a,b),(a,c),(a,d),(b,a),(b,c),(b,d),(c,a),(c,b),(c,d),(d,a),(d,b),(d,c)

2. Broj neuređenih izbora (kombinacija):

Da bismo dobili broj neuređenih izbora (kombinacija), koristimo formulu za kombinacije:

$C(4,2) = P(4,2)/2! = 12/2 = 6$

Neuređeni izbori su:

{a,b},{a,c},{a,d},{b,c},{b,d},{c,d}

Objašnjenje:

Broj uređenih izbora (permutacija) je veći jer uzima u obzir redosled elemenata. Međutim, broj neuređenih izbora (kombinacija) je manji jer se ne računa redosled — npr. (a,b) i (b,a) su isti u slučaju neuređenih izbora i broje se samo jednom.

Kombinacije sa i bez ponavljanja

Kombinacije su neuređeni izbori elemenata iz skupa, ali mogu biti sa ili bez ponavljanja:

1. Kombinacije bez ponavljanja:

a. Kombinacije bez ponavljanja su podskupovi skupa u kojima svaki element može biti izabran samo jednom.
b. Formula za broj kombinacija bez ponavljanja je:

$$C(n,m) = \binom{n}{m} = \frac{n!}{m!(n-m)!}$$

Ovde, n je broj elemenata u skupu, a m je broj elemenata koji biramo.

Primer: Ako imamo skup $S=\{1,2,3,4\}$ i želimo da izaberemo 2 elementa, moguće kombinacije su $\{1,2\},\{1,3\},\{1,4\},\{2,3\},\{2,4\},\{3,4\}$. Broj ovih kombinacija je:

$$C(4,2) = \frac{4!}{2!(4-2)!} = 6$$

Kombinacije sa ponavljanjem:

- Kod kombinacija sa ponavljanjem, svaki element iz skupa može biti izabran više puta.
- Formula za broj kombinacija sa ponavljanjem je:

$$C(n+m-1,m) = \frac{(n+m-1)!}{m!(n-1)!}$$

Primer: Ako imamo skup $S=\{a,b,c\}$ i biramo 2 elementa sa ponavljanjem, kombinacije su $\{a,a\},\{a,b\},\{a,c\},\{b,b\},\{b,c\},\{c,c\}$. Broj ovih kombinacija je:

$$C(3+2-1,2) = \frac{4!}{2!2!} = 6$$

Algoritam za generisanje kombinacija bez ponavljanja:

Kombinacije bez ponavljanja mogu se generisati rekurzivno.

```python
def generate_combinations(skup, m):
    if m == 0:
        return [[]]
    if len(skup) < m:
        return []

    prvi = skup[0]
    ostatak_skupa = skup[1:]

    kombinacije = generate_combinations(ostatak_skupa, m - 1)
    for k in kombinacije:
        k.insert(0, prvi)

    kombinacije_rekurzivni_poziv = generate_combinations(ostatak_skupa, m)

    return kombinacije + kombinacije_rekurzivni_poziv

```

Algoritam za generisanje kombinacija sa ponavljanjem:

```python
def generate_combinations_with_repetition(skup, m):
    if m == 0:
        return [[]]

    kombinacije = []
    for i in range(len(skup)):
        trenutni = skup[i]
        for k in generate_combinations_with_repetition(skup[i:], m - 1):
            kombinacije.append([trenutni] + k)

    return kombinacije
```

Kombinacije sa i bez ponavljanja imaju značajne primene u raznim oblastima računarstva i programiranja:

1. **Optimizacija**: Kombinacije se koriste u rešavanju problema optimizacije kao što su "knapsack problem" i "subsetsum". U ovim slučajevima, potrebno je generisati sve moguće podskupove skupa opcija da bi se pronašlo optimalno rešenje. Na primer, u "knapsack" problemu biraju se podskupovi predmeta tako da njihova ukupna težina ne prelazi kapacitet ranca, dok vrednost izabranih predmeta bude maksimalna.
2. **Testiranje softvera**: Kombinatorno testiranje koristi m-kombinacije za kreiranje seta testova koji pokrivaju sve moguće kombinacije ulaznih podataka. Ovo je posebno korisno u automatizovanom testiranju, gde je potrebno generisati sve kombinacije ulaza kako bi se osiguralo da softver funkcioniše ispravno u svim mogućim scenarijima.

3. **Kriptografija**: Kombinacije se često koriste za generisanje mogućih ključeva za šifrovanje i dešifrovanje podataka. Kriptografi analiziraju otpornost algoritama na napade tako što procenjuju sve moguće kombinacije ključnih vrednosti koje bi potencijalni napadač mogao koristiti u pokušaju kompromitovanja sistema.
4. **Analiza podataka**: U velikim skupovima podataka, kombinacije omogućavaju istraživanje podskupova i odnosa među podacima. Na primer, analitičari mogu generisati različite kombinacije karakteristika (feature subsets) kako bi pronašli optimalnu grupu za treniranje modela mašinskog učenja.

Određivanje broja monotonih neopadajućih konačnih nizova brojeva pomoću kombinacija multiskupa

Određivanje broja monotonih neopadajućih konačnih nizova brojeva pomoću kombinacija multiskupa odnosi se na problem pronalaženja broja načina na koje se može rasporediti određeni broj objekata u određeni broj grupa, gdje su objekti identični, a grupe mogu biti ponovljene.

U matematici, problem se svodi na određivanje broja raspodjele *n* identičnih objekata u *k* grupa, što je poznato kao *kombinacija sa ponavljanjem*. Ovaj broj se može odrediti pomoću formule:

$$\binom{n+k-1}{k-1}$$

Gdje:

- *n* predstavlja broj objekata (broj elemenata niza).
- *k* predstavlja broj grupa (različitih vrijednosti koje niz može da ima).

Pošto je monotono neopadajući niz objekata takav da elementi mogu da se ponavljaju ali su sortirani kao npr. niz [1,2,2,18,29,30,30], to znači da želimo da nađemo broj različitih načina na koje možemo rasporediti *n* elemenata tako da ostanu sortirani.

Primjer

Ako želimo da odredimo broj monotonih neopadajućih nizova od 5 elemenata sa vrijednostima u skupu

{1, 2, 3}, tada je:

- n=5
- k=3

Broj takvih nizova je:

$$\binom{5+3-1}{3-1} = \binom{7}{2} = \frac{7!}{5! * 2!}$$

Dakle, postoji 21 različit način da se formira monoton neopadajući niz od 5 elemenata sa vreijdnostima iz skupa {1, 2, 3}.

Zadatak: Ako imamo kutiju sa 7 kuglica koje mogu biti crvene, plave ili žute boje. Na koliko različitih načina možemo rasporediti kuglice u redoslijed tako da boje ostaju monotono raspoređene (na primjer, prvo crvene, zatim plave, pa žute)?

Rješenje: Bitno je da zbir kuglica bude 7, dakle ***c + p + ž = 7***

Dalje, koristićemo kombinacije sa ponavljanjem, obzirom da su u pitanju identični objekti(kuglice) koje raspoređujemo u 3 grupe tj. 3 boje.

$$\binom{7+3-1}{3-1} = \binom{9}{2} = \frac{9!}{7! * 2!}$$

Algoritam i program za generisanje m-kombinacija multuskupa

Da bismo generisali m-kombinacije multuskupa, možemo koristiti rekurzivni pristup. M-kombinacije multuskupa su svi mogući načini odabira m elemenata iz skupa koji može sadržavati duplikate.

```python
def generate_multiset_combinations(multiset, m, start=0, current_combination=[], all_combinations=[]):
    # Baza slučaja: ako je trenutna kombinacija dužine m, dodaj je u sve kombinacije
    if len(current_combination) == m:
        all_combinations.append(current_combination.copy())
        return
    # Generisanje kombinacija
    for i in range(start, len(multiset)):
        current_combination.append(multiset[i])
        generate_multiset_combinations(multiset, m, i, current_combination, all_combinations)
        current_combination.pop()  # Uklanjanje poslednjeg dodatog elementa
# Primer upotrebe
multiset = [1, 2, 2, 3]  # Multiskup sa duplikatima
m = 2  # Broj elemenata u kombinaciji
all_combinations = []
generate_multiset_combinations(multiset, m, all_combinations=all_combinations)
# Ispis rezultata
for combination in all_combinations:
    print(combination)
```

```
Run    main
[1, 1]
[1, 2]
[1, 2]
[1, 3]
[2, 2]
[2, 2]
[2, 3]
[2, 2]
[2, 3]
[3, 3]
```

Ovaj algoritam koristi rekurzivni pristup. Počinje sa praznom listom i dodaje elemente iz multiskupa jedan po jedan. Kada trenutna kombinacija dostigne dužinu m, dodaje se u listu svih kombinacija. U svakom koraku, algoritam istražuje sve mogućnosti, uključujući duplikate, tako da se generišu sve moguće kombinacije.