

Praktikum 3

Kombinatorika. Rekursioon (I)

Kombinatsioonide mitterekursiivselt leidmise idee

Kombinatsioonide rekursiivselt leidmise idee

Permutatsioonide rekursiivselt leidmise idee

Variatsioonid

Kombinatorsete valimite genereerimine

Indeksite kombinatsioonide generaator (Java)

Indeksite permutatsioonide generaator (Java)

Harjutusülesanded

- **KMB_1. Kombinatsioonid mitterekursiivselt**
- **KMB_2. Permutatsioonid rekursiivselt**

Kombinatsioonide mitterekursiivselt leidmise idee



Iga kombinatsioon n -elemendilistest järjestist k kaupa on määratud järjendi ühe maskiga. Maskideks on bitijärjendid pikkusega n , milles igas k ühte. Näiteks maskid $n = 5$ ja $k = 3$ korral:

00111
01011
01101
01110
10011
10101
10110
11001
11010
11100

A B C D E

järjend

1 0 1 1 0

mask

A C D

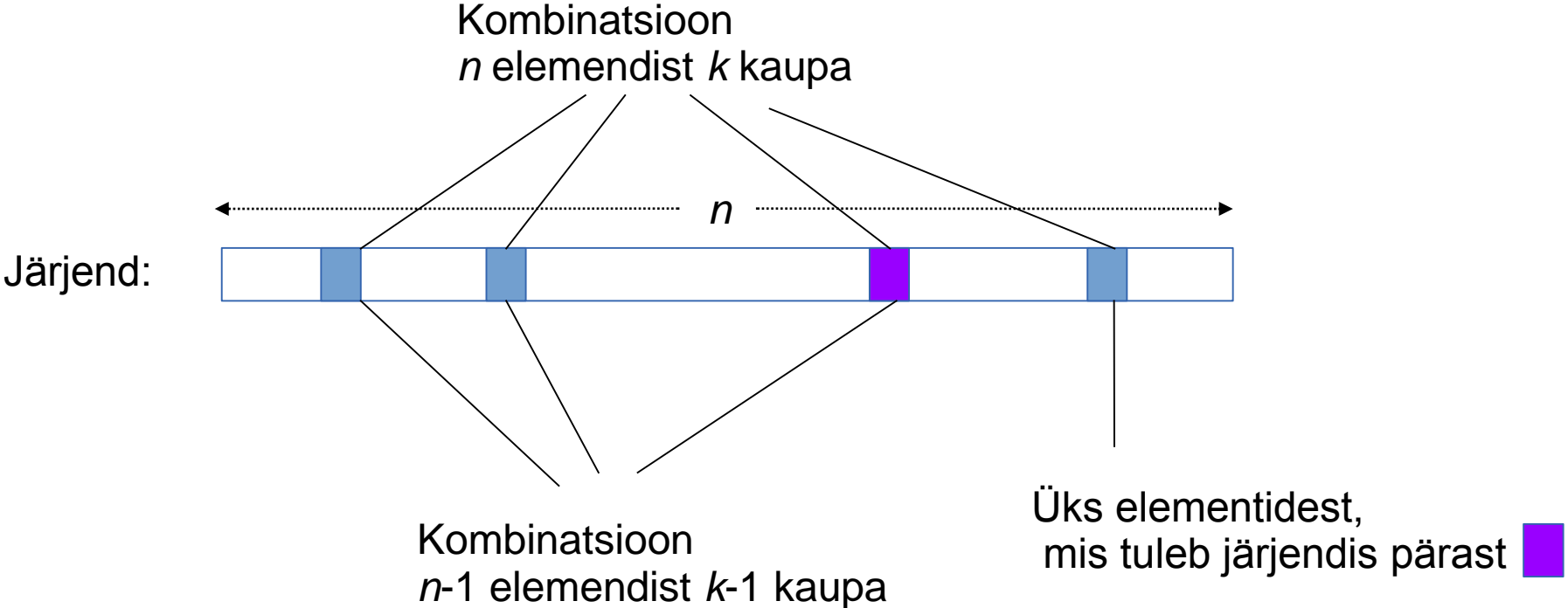
maskile vastav kombinatsioon (5-st 3 kaupa)

Kõikvõimalikud bitijärjendid pikkusega n on arvude

$0, 1, 2, \dots, 2^n - 1$

kahendkujude n viimast numbrit; $n = 5$ korral siis 00000, 00001, 00010, ..., 11111.

Kombinatsioonide rekursiivselt leidmise idee



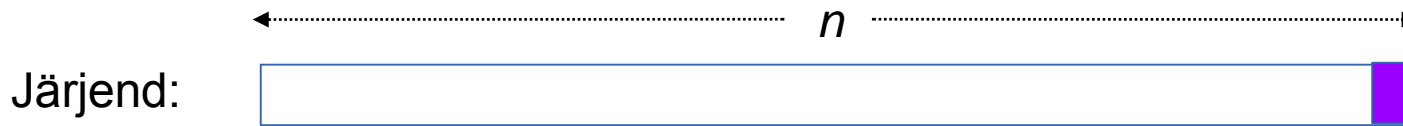
See kombinatsioon
7 elemendist 3 kaupa (B C E): A B C D E F G H

annab

3 kombinatsiooni
8 elemendist 4 kaupa:

B C E F
B C E G
B C E H

Permutatsioonide rekursiivselt leidmise idee



Iga permutatsioon n elemendist on saadav sel teel, et mingile permutatsioonile $n-1$ (esimesest) elemendist on mingisse positsiooni lisatud järjendi viimane element.

Näiteks 6-liikmelise järjendi

A B C D E F

esimese 5 liikme üks permutatsioone on B A E D C

Sellest saame järgmised antud 6-liikmelise järjendi permutatsioonid:

F B A E D C
B F A E D C
B A F E D C
B A E F D C
B A E D F C
B A E D C F

Variatsioonid

Variatsioonideks järjendi elementidest k kaupa on selle järjendi kõigi k -kaupa kombinatsioonidest saadavad permutatsioonid.

Kombinatorsete valimite genereerimine

Olgu lähtehulgaks n -elemendiline indekseeritud andmekogum (massiiv, list, sõne vmt). Sellest kombinatorsete valimite (kombinatsioonide, permutatsioonide, variatsioonide) genereerimiseks on otstarbekohane kasutada indeksite $(0, 1, \dots, n-1)$ komplektide generaatorit. Viimane ei sõltu lähtehulga ega selle elementide tüübist.

Kombinatsioonide n elemendist k kaupa leidmiseks sobib generaator, mis annab välja indeksite $(0, 1, \dots, n-1)$ kombinatsioone k kaupa. Näiteks $n=5$, $k=3$ korral genereeritakse komplektid $\{2, 3, 4\}$ $\{1, 3, 4\}$ $\{1, 2, 4\}$ $\{1, 2, 3\}$ $\{0, 3, 4\}$ $\{0, 2, 4\}$ $\{0, 2, 3\}$ $\{0, 1, 4\}$ $\{0, 1, 3\}$ $\{0, 1, 2\}$. Kui komplekti tüübiks (Javas) on $int[]$, siis järjekordsele komplektile $komp$ vastavaks kombinatsiooniks (valimiks) lähtehulga L elementidest on

$L[komp[0]], L[komp[1]], L[komp[2]]$ – kui L on massiiv

ja

$L.get(komp[0]), L.get(komp[1]), L.get(komp[2])$ – kui L on *ArrayList*

ja

$L.charAt(komp[0]), L.charAt(komp[1]), L.charAt(komp[2])$ – kui L on sõne (*String*).

Permutatsioonide n elemendist leidmiseks sobib generaator, mis annab välja indeksite $(0, 1, \dots, n-1)$ permutatsioone.

Indeksite kombinatsioonide generaator (Java)

Antud juhul on generaatoriks klass, mille isendi konstrueerimisel antakse parameetritena arv n – mitmest indeksist, ja arv k – mitme kaupa genereerida.

Klass sisaldab avalikku isendimeetodit *next()* tagastustüübiga *int[]*, mille järjekordsel rakendamisel tagastatakse järjekordne indeksite komplekt – üks kombinatsioon k -kaupa arvudest $0, 1, \dots, n-1$ (või tagastatakse *null*, kui kombinatsioone rohkem ei ole).

Indeksite permutatsioonide generaator (Java)

Antud juhul on generaatoriks klass, mille isendi konstrueerimisel antakse parameetrina arv n – mitmest indeksist.

Klass sisaldab avalikku isendimeetodit *next()* tagastustüübiga *int[]*, mille järjekordsel rakendamisel tagastatakse järjekordne indeksite komplekt – üks permutatsioon arvudest $0, 1, \dots, n-1$ (või tagastatakse *null*, kui permutatsioone rohkem ei ole).

Harjutusülesanded

KMB_1. Kombinatsioonid mitterekursiivselt

Programmeerida mitterekursiivne generaator-klass indeksite hulgast $\{0, 1, 2, \dots, n-1\}$ k kaupa kombinatsioonide leidmiseks. Vt *Gen_Komblnd_MALL.java*.

Java

Ühtede arv täisarvu m kahendkujus:
`Integer.bitCount(m)`

Kui m on indeksite järjendi $\{0, 1, 2, \dots, n-1\}$ üks maskidest, siis indeks i kuulub selle maskiga määratud valikusse, kui Java avaldis

$$((m \gg (n-1-i)) \% 2 == 1)$$

on tõene.

Järjend $\{0, 1, 2, \dots, n-1\}$ on virtuaalne, seda ei tule konstrueerida.

KMB_2. Permutatsioonid rekursiivselt

Programmeerida rekursiivne generaator-klass indeksite hulga $\{0, 1, 2, \dots, n-1\}$ permutatsioonide leidmiseks. Vt ka *Gen_KomblndRek.java*.