**Imię i nazwisko:** Jakub Pranica **Grupa:** Informatyka -13i **Data:** 13-03-2017 **Przedmiot:** Metody Programowania

**Wykładowca:** dr inż. Zbigniew Kokosiński **Semestr:** 2

**Temat:** Napisać algorytm generujący co najwyżej lub dokładnie k-blokowe podziały zbioru n-elementowego {1,...,n}

**Rozbicie zbioru** (inaczej: podział zbioru, partycja zbioru) – dla niepustego zbioru A to taka rodzina II niepustych podzbiorów tego zbioru, że każdy element zbioru A należy do dokładnie jednego podzbioru tej rodziny.

Liczba sposobów podziału skończonego zbioru n-elementowego wyraża się n-tą liczbą Bella.

**Liczba Bella** jest to liczba wszystkich co najwyżej n-blokowych podziałów danego zbioru n-elementowego.

Przykładowo:

B0 = 1 ponieważ zbiór 0-elementowy {} ma jeden podział: {}

B1 = 1 ponieważ zbiór 1-elementowy {1} ma jeden podział: {1}

B2 = 2 ponieważ zbiór 2-elementowy {1,2} ma dwa podziały: {1,2} oraz {1},{2}

**Opis programu:** Program prosi użytkownika o wprowadzenie liczby n oraz liczby k. Gdzie n oznacza ilość elementów rozważanego zbioru, natomiast k określa liczbę bloków. Użytkownik wybiera również pomiędzy dwoma dostępnymi wariantami - generacją co najwyżej k-blokowych lub dokładnie k-blokowych podziałów. Następnie program generuje zbiór n-elementowy złożony z samych jedynek {1,1,...,1} i zapisuje go w tablicy - a. Tablica a będzie zawierać kolejne reprezentacje podziałów zbioru. Po inicjalizacji następuje sprawdzenie czy tablica a zawiera reprezentacje podziału zgodną z warunkami wybranego wariantu. Jeżeli tak, to wypisuje ją na ekran lub do pliku. Po wypisaniu tablica a przekształcana jest tak aby reprezentowała kolejny podział omawianego zbioru {1,...,n}. Program wykonuje tę czynność dopóki nie uzyska wszystkich możliwych podziałów. Na koniec podaje dodatkowo ich ilość. **(Pseudokod na odwrocie kartki)**

**Napotkane problemy: 1.**Reprezentacja podziałów zbioru, za pomocą tablic w języku programowania c++ **2.**Wygenerowanie wszystkich wartości funkcji wyboru (wszystkich możliwych podziałów o wybranych właściwościach) **3.** Sprawdzenie czy wygenerowany podział spełnia warunki wariantu wybranego przez użytkownika **4.**Wypisanie na ekran i do pliku wszystkich reprezentacji podziałów zbioru zapisanych w postaci tablicy.

**Przykładowy wynik programu:**

Program generuje podziały zbioru n-elementowego. {1,...,n} Gdzie wypisać wyniki? 1 - na ekran 2 - do pliku **1**

Podaj n: **3** Podaj t: **4**

Warianty:

1. Generacja co najwyżej k-blokowych podziałów (1<=k<=n)

2. Generacja dokładnie k-blokowych podziałów (1<=k<=n)

Wybierz wariant: **1** Podaj n: **4** Podaj k: **4**

Rozpoczynam wypisywanie kolejnych możliwych co najwyżej k-blokowych podziałów wybranego zbioru.

1. 1111

2. 1112

3. 1121

4. 1122

5. 1123

6. 1211

7. 1212

8. 1213

9. 1221

10. 1222

11. 1223

12. 1231

13. 1232

14. 1233

15. 1234

Ilość wygenerowanych podziałów: 15

Wybierz wariant: **2** Podaj n: **6** Podaj k: **3**

Rozpoczynam wypisywanie kolejnych możliwych dokładnie k-blokowych podziałów wybranego zbioru.

1. 11123

2. 11213

3. 11223

4. 11231

5. 11232

6. 11233

7. 12113

8. 12123

9. 12131

10. 12132

11. 12133

12. 12213

13. 12223

14. 12231

15. 12232

16. 12233

17. 12311

18. 12312

19. 12313

20. 12321

21. 12322

22. 12323

23. 12331

24. 12332

25. 12333

Ilość wygenerowanych podziałów: 25

**Pseudokod:**

**Wypisz na ekran:** " Program generuje podziały zbioru n-elementowego. {1,...,n} \n"

**Użytkownik podaje liczbę**, a program przypisuje ją do zmiennej **n; //**ilość elementów rozważanego zbioru

**Użytkownik podaje liczbę**, a program przypisuje ją do zmiennej **k; //**liczba bloków

**Zdefiniuj tablicę typu int a i wypełnij ją jedynkami:**

a = {1,1,...,1} //Tablica a zawiera zbiór, który jest reprezentacją kolejnych podziałów omawianego zbioru {1,...,n}

**Zdefiniuj** bool czyDoklanieKblokowe = false;

**Wypisz na ekran:** "Warianty: \n "

"1. Generacja co najwyżej k-blokowych podziałów (1<=k<=n) \n"

"2. Generacja dokładnie k-blokowych podziałów (1<=k<=n) \n"

**Użytkownik wybiera wariant. Jeżeli wybrał 2-gi wariant to ustaw** czyDoklanieKblokowe **na true.**

**//** if(wariant==2) czyDoklanieKblokowe = true; else czyDoklanieKblokowe = false;

**Wypisz na ekran:** "Rozpoczynam wypisywanie kolejnych możliwych podziałów wybranego zbioru. \n "

**Zdefiniuj** int ilosc = 0; **oraz** bool czyWypisac = true;

**Powtarzaj:**

**Jeżeli** czyDoklanieKblokowe == true **to:**

**Sprawdź czy największa wartość tablicy a jest równa k, jeżeli tak to ustaw czyWypisac na true, natomiast w przeciwnym wypadku ustaw czyWypisac na false.**

**W przeciwnym wypadku od razu ustaw czyWypisac na true.**

**Jeżeli** czyWypisac == true **to:**

**Zwiększ** ilosc **o 1 (**ilosc++;**)**

**Wypisz na ekran lub do pliku:** ilosc + ". \***tu zawartość tablicy a**\* \n" //tablica a zawiera reprezentację podziału wybranego zbioru {1,...,n}

**Zwiększ ostatni element tablicy a o 1 (**a[n-1]++;**)**

**Ustaw tablicę a tak aby przedstawiała kolejną wartość funkcji wyboru (kolejny podział zbioru):**

for(int i = 1; n-i-1>=0 ; ){

m = 0;

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

if( a[j] > m ){

m = a[j];

}

}

if( a[n-i] == m+2 || a[n-i] == k+1 ){

a[n-i-1]++;

a[n-i] = 1;

i++;

} else break;

}

**dopóki** a[0] == 1; //dopóki nie wygenerujesz wszystkich podziałów

**Wypisz na ekran lub do pliku:** "Ilość wygenerowanych podziałów:" ilosc**;**

**Pełny kod programu w C++ dostępny pod adresem:** http://pastebin.com/4kdfLYTB

**Wnioski:** Program działa i prawidłowo generuje wszystkie możliwe co najwyżej lub dokładnie k-blokowe podziały zbioru n-elementowego {1,...,n} dla podanych przez użytkownika danych, takich jak: Ilość elementów zbioru (n) oraz liczbę bloków (k). Czas wygenerowania oraz wypisania wszystkich sekwencji rośnie wraz ze wzrostem wartości n.

Liczba Bella jest to liczba wszystkich co najwyżej n-blokowych podziałów danego zbioru n-elementowego. Program pozwolił na uzyskanie następujących wyników: B(2)=2 B(3)=5 B(4)=15 B(5)=52

**Literatura:**

* pl.wikipedia.org/wiki/Rozbicie\_zbioru oraz pl.wikipedia.org/wiki/Liczba\_Bella
* http://edu.pjwstk.edu.pl/wyklady/
* http://www.student.krk.pl/024-Dudek-Krakow/ - Strona poświęcona kombinatoryce - Rafał Dudek
* "Kombinatoryka dla programistów" - Lipski Witold