

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА
ПОЛІТЕХНІКА”**

Кафедра систем штучного інтелекту

Лабораторна робота
з дисципліни
«Дискретна математика»

Виконав:
студент групи КН-109
Коваль Назар
Викладач:
Мельникова Н.І.

Львів – 2018 р.

Лабораторна робота № 6

Тема: Генерація комбінаторних конфігурацій

Мета роботи: набути практичних вмінь та навичок при комп'ютерній реалізації комбінаторних задач.

Завдання № 1. Використовуючи теоретичні відомості, розв'язати наступні комбінаторні задачі за своїм варіантом:

Варіант № 8

1. З букв розрізаної абетки складено слово «конус». Скільки «слів» можна отримати, якщо переставляти букви у цьому слові?
 2. Скільки різних чотирицифрових чисел можна скласти з цифр 1, 2, 3, 4, 5, 6, 7 так, щоб у кожному з них була цифра 1? (Цифри в числі не повинні повторюватися).
 3. Із групи до складу якої входять 8 хлопчиків і 3 дівчинки, треба сформувати команду з 6 чоловік. Скільки існує способів формування такої команди?
 4. Скільки можна скласти різних неправильних нескоротних дробів, чисельниками і знаменниками яких є числа 3, 5, 7, 9, 11, 13, 17, 27?
 5. Скількома способами можна переставити букви в слові «обороздатність», щоб дві букви «о» не стояли поряд?
 6. П'ять учнів мають підготувати 10 докладів на семінар (кожен по два). Скількома способами вони можуть розподілити доклади між собою?
 7. Підкидаються три гральні кістки. Скільки може бути варіантів таких, щоб не виповнилась жодна умова: 1) на всіх кістках випали трійки; 2) на всіх кістках випали попарно різні числа; 3) рівно на одній з них випала одиниця?
-
1. У слові «конус» 5 букв. Кількість «слів», що можна утворити – кількість перестановок. Відповідь: $5! = 120$
 2. Якщо забрати 1 і шукати кількість 3-цифрових чисел, що можна скласти з 6 цифр, а потім додати 1 в одну із 4 позицій, то маємо: $4 * A_6^3 = 480$
 3. Якщо всього у нас 11 людей і нам не важливий порядок у якому вони йдуть, то кількість вибірок по 6 чоловік дорівнює $C_{11}^6 = 462$
 4. Для того щоб дати відповідь на цю задачу не потрібно використовувати стандартні формули комбінаторики. Із числом 3 можливо скласти 5 таких дробів, із 5-кою – 6, 7 – 5, 9 – 3, 11 – 3, 13 – 2, 17 – 1. Відповідь:
 $5+6+5+3+3+2+1=25$
-
5. Забравши всі букви “о” зі слова, кількість перестановок буде рівна 12! Розмістити букви “о” за умовою можна за допомогою формули A_{13}^4
Отже кількість способів буде рівна $12! * A_{13}^4$

6. Перший учень повинен вибрати 2 доклади з десяти. Це можна зробити C_{10}^2 разів. Наступий відповідно C_8^2 разів. Загальна кількість способів буде рівна $C_{10}^2 * C_8^2 * C_6^2 * C_4^2 * C_2^2 = 113400$.

7. Згідно другої умови кількість варіантів випадання рівна 108. Перша умова може виконатись лише 1 раз, отже кількість варіантів випадання згідно двох умов буде рівна $108 - 1 = 107$. Згідно першої і другої умови, третя умова може виконатись 15 разів. Загальна кількість варіантів випадання згідно усіх умов буде рівна $107 - 15 = 92$.

Завдання №2. Запрограмувати за варіантом обчислення кількості розміщення (перестановок, комбінацій, алгоритму визначення наступної лексикографічної сполуки, перестановки) та формулу Ньютона і побудувати за допомогою неї розклад за варіантом

Варіант № 8

Розташувати наведені перестановки елементів множини {1, 2, 3, 4, 5, 6} у лексикографічному порядку 234561, 231456, 165432, 156423, 543216, 541236, 231465, 314562, 432561, 654321, 654312, 435612. Побудувати розклад $(x + y)^8$.

Код програми:

```
#include<stdio.h>
#include<stdlib.h>
void qsort_column(int *a, int first, int last) {
    int i = first, j = last, m = a[(first + last) / 2], swap;
    do {
        while (a[i] < m) i++;
        while (a[j] > m) j--;
        if (i <= j) {
            if (i < j) {
                swap = a[i];
                a[i] = a[j];
                a[j] = swap;
            }
            i++;
            j--;
        }
    } while (i <= j);
    if (i < last)
        qsort_column(a, i, last);
    if (first < j)
        qsort_column(a, first, j);
}
int fact(int n)
{
    int fact = 1;
    if (n < 2) {
        return 1;
    }
    for (int i=1; i <= n; i++) {
        fact *= i;
    }
    return fact;
}
```

```

int combination(int n, int m) {
    return fact(n) / (fact(n - m) * fact(m));
}

int main()
{
    int array[12]={234561, 231456, 165432, 156423, 543216, 541236, 231465, 314562, 432561, 654321,
654312, 435612};
    qsort_column(array,0,11);
    printf("Leks array:\n");
    for(int i=0; i<12; i++)
    {
        printf("%d",array[i]);
        printf(" ");
    }
    printf("\n");
    for(int i=0; i<=8;i++)
    {
        printf("%d*x^%d*y^%d\n",combination(8,i),i,8-i);
    }
    printf("\n");
}

```

Результат роботи програми:

```

Leks array:
156423|165432|231456|231465|234561|314562|432561|435612|541236|543216|654312|654321|
1*x^0*y^8
8*x^1*y^7
28*x^2*y^6
56*x^3*y^5
70*x^4*y^4
56*x^5*y^3
28*x^6*y^2
8*x^7*y^1
1*x^8*y^0
jharvard@appliance (~/Desktop):

```

Висновок: на лабораторній роботі ми набули практичних вмінь та навичок при комп'ютерній реалізації комбінаторних задач.