**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ**

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧЕРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»**

**(БГТУ им.В.Г.Шухова)**

Кафедра программного обеспечения вычислительной техники и

автоматизированных систем

Лабораторная работа №2.1

дисциплина: Дискретная математика

тема: «Комбинаторные объекты»

Выполнил: ст. группы ПВ-201

Машуров Дмитрий Русланович

Проверил: Бондаренко Т.В.

Белгород 2021

**Лабораторная работа № 2.1**

**Алгоритмы порождения комбинаторных объектов**

**Цель работы:** изучить основные комбинаторные объекты, алгоритмы их порождения, программно реализовать и оценить временную сложность алгоритмов.

**Задания**

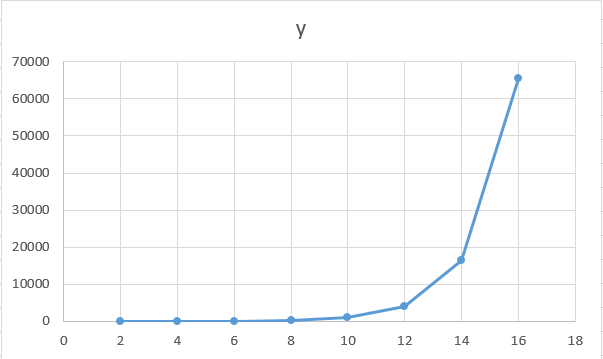
1. Реализовать алгоритм порождения подмножеств.
2. Построить график зависимости количества всех подмножеств от мощности множества.
3. Построить графики зависимости времени выполнения алгоритмов п.1 на вашей ЭВМ от мощности множества.
4. Определить максимальную мощность множества, для которого можно получить все подмножества не более чем за час, сутки, месяц, год на вашей ЭВМ.
5. Определить максимальную мощность множества, для которого можно получить все подмножества не более чем за час, сутки, месяц, год на ЭВМ, в 10 и в 100 раз быстрее вашей.
6. Реализовать алгоритм порождения сочетаний.
7. Построить графики зависимости количества всех сочетаний из n по k от k при n = (5, 7, 9).
8. Реализовать алгоритм порождения перестановок.
9. Построить график зависимости количества всех перестановок от мощности множества.
10. Построить графики зависимости времени выполнения алгоритма п.8 на вашей ЭВМ от мощности множества.
11. Определить максимальную мощность множества, для которого можно получить все перестановки не более чем за час, сутки, месяц, год на вашей ЭВМ.
12. Определить максимальную мощность множества, для которого можно получить все перестановки не более чем за час, сутки, месяц, год на ЭВМ, в 10 и в 100 раз быстрее вашей.
13. Реализовать алгоритм порождения размещений. 14. Построить графики зависимости количества всех размещений из n по k от k при n = (5, 7, 9)

**Выполнение:**

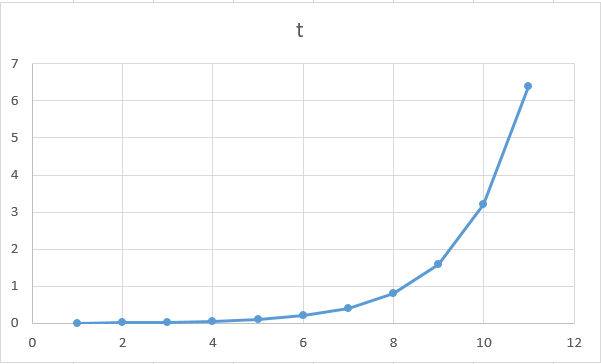
1. Реализация подмножеств:

static void get\_subsets\_inner(int \*a, int n, int i) {  
 static int d[100];  
  
 for (int x = 0; x < 2; ++x) {  
 d[i] = x;  
  
 if (i == n - 1) {  
 output\_array\_with\_mask(a, n, d);  
 } else {  
 get\_subsets\_inner(a, n, i + 1);  
 }  
 }  
}

1. График



1. График зависимости выполнения алгоритма на моей ЭВМ



При мощности множества 8 затрачивается сек, следовательно:

Тогда:

1. Определю максимальную мощность множества, для которого можно получить все подмножества не более чем:

* За час:
* За сутки:
* За месяц:
* За год:

1. Определю максимальную мощность множества, для которого можно получить все подмножества не более чем за час, сутки, месяц, год на ЭВМ, в 10 и в 100 раз моей

Получу формулу:

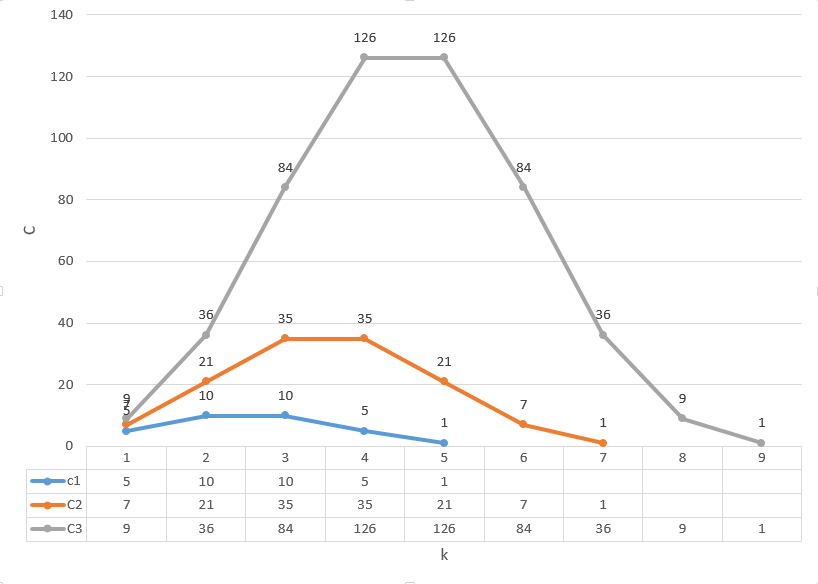
Соответственно:

* За час:
  + В 10 раз:
  + В 100 раз:
* За сутки:
  + В 10 раз:
  + В 100 раз:
* За месяц:
  + В 10 раз:
  + В 100 раз:
* За год:
  + В 10 раз:
  + В 100 раз:

1. Реализовать алгоритм порождения сочетаний

void get\_combinations\_inner(int a[], int n, int i, int k, int b) {  
 static int c[100];  
  
 for (int j = b; j < n; ++j) {  
 c[i] = a[j];  
  
 if (i == k - 1) {  
 output\_arr(c, k);  
 } else {  
 get\_combinations\_inner(a, n, i + 1, k, j + 1);  
 }  
 }  
}

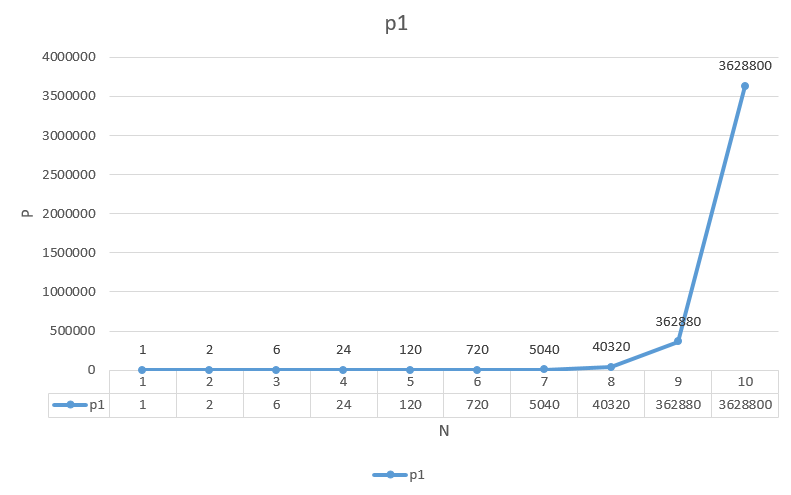
1. Построить графики зависимости количества всех сочетаний из n по k от k при n = (5, 7, 9)



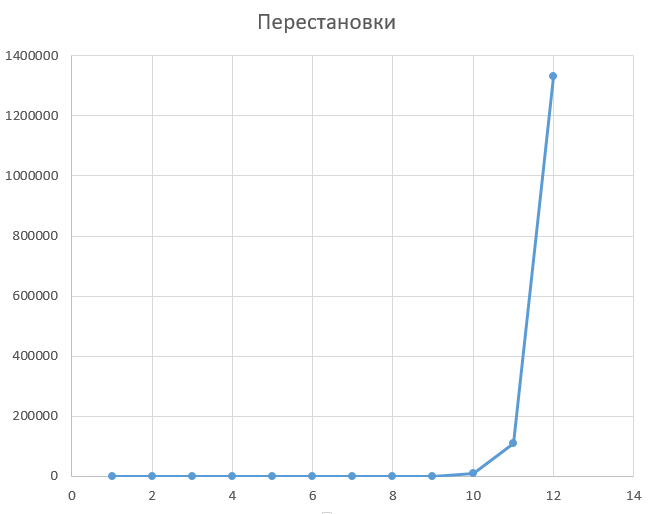
1. Реализовать алгоритм порождения перестановок

static void get\_permutations\_inner(int a[], int n, int i) {  
 static int m[100];  
 static int t[100];  
  
 for (int j = 0; j < n; ++j) {  
 if (t[j] == 0) {  
 m[i] = a[j];  
 t[j] = 1;  
  
 if (i == n - 1) {  
 output\_arr(m, n);  
 } else {  
 get\_permutations\_inner(a, n, i + 1);  
 }  
  
 t[j] = 0;  
 }  
 }  
}

1. Построить график зависимости количества всех перестановок от мощности множества



1. Построить графики зависимости времени выполнения алгоритма п.8 на вашей ЭВМ от мощности множества.



Определю формулу для вычисления времени :

Следовательно:

1. Определить максимальную мощность множества, для которого можно получить все перестановки не более чем за час, сутки, месяц, год на вашей ЭВМ.

* Не более чем за час: 10
* Не более чем за сутки: 11
* Не более чем за месяц: 13
* Не более чем за год: 14

1. Определить максимальную мощность множества, для которого можно получить все перестановки не более чем за час, сутки, месяц, год на ЭВМ, в 10 и в 100 раз быстрее вашей.

Выражу формулу для вычисления:

Соответственно:

* Не более чем за час
  + В 10 раз мощнее
  + В 100 раз мощнее
* Не более чем за сутки
  + В 10 раз мощнее
  + В 100 раз мощнее
* Не более чем за месяц
  + В 10 раз мощнее
  + В 100 раз мощнее
* Не более чем за год
  + В 10 раз мощнее
  + В 100 раз мощнее

1. Реализовать алгоритм порождения размещений. 14. Построить графики зависимости количества всех размещений из n по k от k при n = (5, 7, 9)

Реализация:

void get\_accommodation\_inner(int a[], int n, int k, int i) {  
 static int o[100];  
 static int t[100];  
  
 for (int j = 0; j < n; ++j) {  
 if (t[j] == 0) {  
 o[i] = a[j];  
 t[j] = 1;  
  
 if (i == k - 1) {  
 output\_arr(o, k);  
 } else {  
 get\_accommodation\_inner(a, n, k, i + 1);  
 }  
  
 t[j] = 0;  
 }  
 }  
}

График:

