**Лабораторная работа № 2. Комбинаторика**

**Цель работы:**

Получение практических навыков решения комбинаторных задач.

**Программа работы:**

1. Изучить теорию.

2. Разработать программу формирования перестановок, сочетаний, размещений.

3. Выполнить вычислительные эксперименты.

**Используемые программно-технические средства:**

1. Персональный компьютер типа IBM PC.

2. Turbo Pascal 7.0.

**Краткая теория:**

Комбинаторикой называют раздел дискретной математики, в котором рассматриваются вопросы, связанные с формированием и подсчетом комбинаций из элементов перестановок, сочетаний, размещений.

Перестановкой из элементов называют комбинации отличающиеся порядком расположения элементов.



Количество перестановок определяется по формуле



Сочетанием из элементов по элементам называются комбинации отличающиеся хотя бы одним элементом.



Количество сочетаний без повторений определяется по формуле:



Размещением без повторений из элементов по называют комбинации, отличающиеся либо элементами, либо порядком расположения элементов.



Количество размещений без повторений определяется по формуле:



Число размещений связано с числом перестановок и сочетаний соотношением:



**Математическая постановка задачи:**

Составить программу формирования перестановок, сочетаний, размещений с выводом результатов на экран дисплея.

**Описание программы:**

Данная программа, написанная на языке Паскаль, начинается с раздела переменных, полный список которых представлен в таблице 1. В основе алгоритма программы лежат три процедуры, каждая из которых отвечает за закрепленную за ней часть программы (см. таблицу 2). Выбор требуемой операции происходит путем использования оператора case.

Работа программы начинается с вывода сообщения о необходимости выбрать операцию для выполнения. Далее требуется ввести из скольки и по сколько элементов будет осуществляться данная операция. Результат выполнения операции выводится на экран.

Таблица 1 - Список идентификаторов переменных

|  |  |  |
| --- | --- | --- |
| Идентификатор | Тип | Применение |
| massiwi1 | massiwi1:massiwi; | Для хранения промежуточных результатов |
| massiwi2 | massiwi2:massiwi; | Для хранения промежуточных результатов |
| iz\_skolki | integer | Из скольки элементов |
| po\_skolko | integer | По сколько элементов |
| i, j, | integer | Для организации циклов |
| Nomer | integer | Хранит номер выбранной операции |
| y | integer | Вспомогательная переменная |

Таблица 2 - Список процедур

|  |  |  |  |
| --- | --- | --- | --- |
| Имя процедуры | Формальные параметры | Вызов процедуры | Применение |
| sochetanye | m, y - целые числа; | sochetanye(m,y:integer); | Операция сочетания |
| perestanovka | m, y - целые числа;  s - массив; | perestanovka(m,y:integer; s:mas); | Операция перестановки |
| razmesheniye | m, y - целые числа; | razmesheniye(m,y:integer; s:mas); | Операция размещения |

**Постановка отдельного примера:**

Рассмотрим все возможные перестановки из 7-ми элементов, сочетания из 6 по 3 элемента и размещения из 7 по 3 элемента.

**Вывод**

В результате всей проделанной работы мы получили практические навыки решения комбинаторных задач, также нами была разработана программа на языке Паскаль, реализующая формирование перестановок, сочетаний и размещений с выводом результатов на экран дисплея.

**Приложение**

**Листинг программы**

uses crt;

label kombinatorika;

type

massiwi=array [1..20] of integer;

var

massiwi1:massiwi;

massiwi2:massiwi;

iz\_skolki, po\_skolko:integer;

i,j:integer;

nomer:integer;

y:integer;

procedure perestanovka(m,y:integer; s:massiwi);

var

j,i:integer;

s1:massiwi;

begin

for i:=1 to m do begin

massiwi1[y]:=s[i];

j:=1;

for y:=1 to m do begin

if s[y]<>s[i] then begin

s1[j]:=s[y];

j:=j+1;

end;

end;

if y=iz\_skolki then begin

for j:=1 to iz\_skolki do write(massiwi1[j]);

writeln;

end else

perestanovka(m-1,y+1,s1);

end;

end;

procedure sochetanye(m,y:integer);

var

j,i:integer;

begin

for i:=1 to m do begin

massiwi1[y]:=i;

if y=po\_skolko then begin

for j:=1 to po\_skolko do write(massiwi1[j]);

writeln;

end else

sochetanye(m,y+1);

end;

end;

procedure razmesheniye(m,y:integer; s:massiwi);

var

j,i:integer;

begin

for i:=1 to m do begin

massiwi1[y]:=i;

if y=po\_skolko then begin

for j:=1 to po\_skolko do write(massiwi1[j]);

writeln;

perestanovka(po\_skolko,1,massiwi2);

end else begin

sochetanye(m,y+1);

perestanovka(po\_skolko,1,massiwi2);

end;

end;

end;

Begin

kombinatorika:clrscr;

for i:=1 to 8 do

writeln;

writeln(' Wi dolgni wibrat neobhodimuy operaciyu:');

writeln('-->> 1. Razmeshenie;');

writeln('-->> 2. Perestanovka; ');

writeln('-->> 3. Sochetanie; ');

writeln('-->> 4. Vihod.');

writeln;

write('-->> Wi wibrali:');

readln(nomer);

case nomer of

1: begin

clrscr;

write('Sochetanye iz=');

readln(iz\_skolki);

write(' po=');

readln(po\_skolko);

writeln;

writeln('Sochetanye:');

sochetanye(iz\_skolki,1);

readln;

goto kombinatorika;

end;

2: begin

clrscr;

write('Perestanovka iz=');

readln(iz\_skolki);

for i:=1 to iz\_skolki do massiwi2[i]:=i;

writeln;

writeln('Perestanovka:');

perestanovka(iz\_skolki,1,massiwi2);

readln;

goto kombinatorika;

end;

3:begin

clrscr;

write('Razmeshenie iz=');

readln(iz\_skolki);

write(' po=');

readln(po\_skolko);

for i:=1 to iz\_skolki do massiwi2[i]:=i;

writeln;

writeln('Razmeshenie:');

razmesheniye(iz\_skolki,1,massiwi2);

readln;

goto kombinatorika;

end;

4: end;

end.