**Лабораторная работа №4**

**Полный перебор**

Решить методом полного p-ичного перебора следующие задачи:

3,11. Сумма чисел

В текстовом файле имеется N (N<=20) натуральных чисел по одному в каждой

строке.

Расставить между этими числами знаки "+" и "-" так, чтобы в результате

вычислений получилось число А, запрашиваемое с клавиатуры. Программа должна находить все варианты расстановки знаков, подсчитывать их количество и выводить результаты в текстовый файл.

Алгоритм:

2-чный перебор (+ и -)

1. Ввод A и N с клавиатуры
2. Создание файла input.txt
3. Заполнение файла input.txt числами с клавиатуры
4. Заполнение целочисленного одномерного динамического массива numbers размером N числами из файла input.txt
5. Создание файла output.txt
6. целочисленная переменная o =0 (количество подходящих комбинаций)
7. Заполнение целочисленного одномерного динамического массива znaki размером N-1 нулями
8. Перебор всех комбинаций: значениями массива znaki могут быть 0 или 1. 0 – означает "-", 1 означает "+". Создается переменная sum, которой присваивается значение 0 элемента массива numbers. Далее перебираются значения массива znaki: соответственно значениям 0 или 1 массива znaki,к sum прибавляется, или вычитается значение соответствующего элемента массива numbers. Если значение sum = значению k, то выражение выводится в файл output.txt. Значение переменной o увеличивается на 1. Далее значения элементов массива znaki изменяются при помощи метода бинарного инкремента (bininc).
9. Вывод комбинаций и значения переменной o в файл output.txt.

#include <iostream>

#include <time.h>

#include <cmath>

#include <fstream>

using namespace std;

void bininc(int znaki[], int N) {

int i = N - 2;

while (i >= 0 && znaki[i] == 1) {

znaki[i] = 0;

i--;

}

if (i >= 0) {

znaki[i] = 1;

}

}

int main() {

int N, num, k, o;

o = 0;

cout << "A="; cin >> k;

cout << "N="; cin >> N;

ofstream file("input.txt");

cout << "Enter " << N << " natural numbers separating with enter:" << endl;

for (int i = 0; i < N; ++i) {

cin >> num;

file << num << endl;

}

file.close();

ifstream f("input.txt");

int \*numbers = new int[N];

for (int i = 0; i < N; i++) {

f >> numbers[i];

}

f.close();

ofstream fo("output.txt");

int \*znaki = new int[N - 1];

for (int i = 0; i < N-1; i++) {

znaki[i] = 0;

}

for (int t = 0; t<int(pow(2, N - 1)); t++) {

int sum = numbers[0];

for (int i = 1; i < N; i++) {

if (znaki[i - 1] == 0) sum -= numbers[i];

if (znaki[i - 1] == 1) sum += numbers[i];

if (sum == k && i == N-1) {

o++;

fo << numbers[0];

for (int j = 1;j < N; j++) {

if (znaki[j - 1] == 0) fo << "-";

else fo << "+";

fo << numbers[j];

}

fo << endl;

}

}

bininc(znaki, N);

}

fo << o;

fo.close();

}



