

РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ 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 №2.2
по дисциплине: Дискретная математика
тема: «Задачи выбора»

Выполнил: ст. группы ПВ-211
Чувилко Илья Романович

Проверили:
Рязанов Юрий Дмитриевич

Белгород 2022 г.

Цель занятия: научиться решать задачи выбора.

Задача: На прямой расположены n равноотстоящих друг от друга узлов. Можно ли в узлах разместить n предметов из n -элементного множества так, чтобы центр тяжести находился в одном из узлов. Вес каждого предмета задан.

```
#include <iostream>
#include "vector"

using namespace std;

void GetPermutations
(
    const vector<int> &setAvailable,
    const vector<int> &setChosen,
    vector<vector<int>> &permutations
) {
    if (!setAvailable.empty()) {
        for (int j = 0; j < setAvailable.size(); j++) {
            vector<int> setAvailableCopy = setAvailable;
            vector<int> setChosenCopy = setChosen;
            setChosenCopy.push_back(setAvailableCopy[j]);
            setAvailableCopy.erase(setAvailableCopy.begin() + j);
            GetPermutations(setAvailableCopy, setChosenCopy,
                           permutations);
        }
    } else
        permutations.push_back(setChosen);
}

void getPermutations
(
    const vector<int> &initialSet,
    vector<vector<int>> &permutations
) {
    vector<int> setChosen;
    GetPermutations(initialSet, setChosen, permutations);
}

int CheckoutPivotIndex(vector<int> &nums) {
    int sum = 0, leftSum = 0;
    for (int x: nums) sum += x;
    for (int i = 0; i < nums.size(); ++i) {
        if (leftSum == sum - leftSum - nums[i]) return i;
        leftSum += nums[i];
    }
    return -1;
}

int main() {
    int n;
    cout << "Enter n:\n";
    cin >> n;
    vector<int> initialSet(n);
    cout << "Enter n elements of the set:\n";
    for (auto &i: initialSet)
        cin >> i;
```

```
vector<vector<int>> permutations;  
getPermutations(initialSet, permutations);  
for (auto &i: permutations) {  
    int pivot = CheckoutPivotIndex(i);  
    if (pivot != -1) {  
        for (auto j: i)  
            cout << j << " ";  
        cout << '\n';  
    }  
}  
  
return 0;  
}
```

Вывод: в ходе лабораторной работы мы научились решать задачи с помощью комбинаторных объектов