

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний інститут
імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 9 з дисципліни
«Алгоритми та структури даних-2.
Структури даних»

«Динамічне програмування»

Варіант 5

Виконав студент

ІП-15, Буяло Дмитро Олександрович
(шифр, прізвище, ім'я, по батькові)

Перевірів

Соколовський Владислав Володимирович
(прізвище, ім'я, по батькові)

Київ 2022

Лабораторна робота 9

Динамічне програмування

Мета – вивчити основи динамічного програмування на прикладі задачі про рюкзак та способи його імплементації.

Індивідуальне завдання

Варіант 5

Завдання

Дано n різних предметів, про які відомі їх розмір, або вага, w_i та вартість v_i . Є рюкзак, в який необхідно покласти ці предмети. Для рюкзака відома його місткість (сумарний розмір, або вага, предметів, що можуть бути розміщені у рюкзаку) — W .

Необхідно відібрати таку множину предметів S серед усіх заданих предметів, що (1) їх сумарна розмірність не перевищує місткість рюкзака W та (2) сумарна вартість предметів в множині S є максимально можливою серед усіх інших множин.

Алгоритм динамічного програмування повинен перебрати підзадачі всіх можливих розмірностей, які визначаються двома змінними:

- i – кількість предметів ($1, \dots, n$)
- x – місткість рюкзака ($1, \dots, W$)

1. Програмна реалізація алгоритму

1.1 Вихідний код

Main.cpp:

```
#include "Functions.h"
#include "Item.h"

int main() {
    int maxWeight = 0, count = 0;

    vector<Item> itemList = input("input.txt", maxWeight, count);
    int** backpack = capacity(itemList, count, maxWeight);

    writer("output.txt", backpack, count, maxWeight);
    cout << "Done!";
}
```

Functions.h:

```
#pragma once
#include <iostream>
#include <string>
#include <fstream>
#include <vector>
#include "item.h"

using namespace std;

vector<Item> input(string, int&, int&);
int** capacity(vector<Item>&, int, int);
void writer(string, int**, int, int);
```

Item.h:

```
#pragma once
#include <iostream>

using namespace std;

class Item {
    int weight;
    int cost;
public:
    int getWeight() const;
    void setWeight(int w);
    int getCost() const;
    void setCost(int c);
};
```

Functions.cpp:

```
#include "Functions.h"

vector<Item> input(string name, int& weight, int& count) {
    int space;
    string line;
    vector<Item> list;
    ifstream inFile(name);
    getline(inFile, line, ' ');
    weight = stoi(line);
    getline(inFile, line);
    count = stoi(line);
    list.resize(count);
    for (int i = 0; i < count; i++) {
        getline(inFile, line);
        space = line.find(' ');
        list[i].setCost(stoi(line.substr(0, space)));
        list[i].setWeight(stoi(line.substr(space, line.length() - space)));
    }

    inFile.close();
    return list;
}

int** capacity(vector<Item>& list, int count, int weight) {
    int w, cost;
    int** arr = new int* [count];
    for (int i = 0; i < count; i++) {
        arr[i] = new int[weight];
        w = list[i].getWeight();
        cost = list[i].getCost();
        for (int j = 0; j < weight; j++) {
            if (i == 0) {
                arr[i][j] = 0;
            }
            else if (j >= w) {
                arr[i][j] = max(arr[i - 1][j], arr[i - 1][j - w] + cost);
            }
            else {
                arr[i][j] = arr[i - 1][j];
            }
        }
    }
    return arr;
}

void writer(string name, int** arr, int count, int weight) {
    ofstream outFile(name);
    outFile << arr[count - 1][weight - 1];
    outFile.close();
}
```

Item.cpp:

```
#include "Item.h"

int Item::getWeight() const {
    return weight;
}

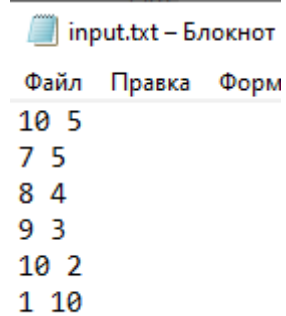
void Item::setWeight(int w) {
    weight = w;
}

int Item::getCost() const {
    return cost;
}

void Item::setCost(int c) {
    cost = c;
}
```

1.2 Приклад роботи

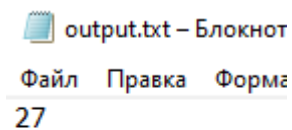
На рисунках 1.1 і 1.2 наведено приклад роботи програми.



input.txt – Блокнот

Файл	Правка	Форм
10	5	
7	5	
8	4	
9	3	
10	2	
1	10	

Рисунок 1.1 – Умова



output.txt – Блокнот

Файл	Правка	Форм
27		

Рисунок 1.2 – Результат роботи програми