

Московский авиационный институт
(национальный исследовательский университет)

Факультет информационных технологий и прикладной
математики

Кафедра вычислительной математики и программирования

Лабораторная работа №2 по курсу «Дискретный анализ»

Студент: Р. Р. Гаптулхаков
Преподаватель: Н. С. Капралов
Группа: М8О-208Б
Дата: 01.05.21
Оценка:
Подпись:

Москва, 2021

Лабораторная работа №6

Задача: Разработать жадный алгоритм решения задачи, определяемой своим вариантом. Доказать его корректность, оценить скорость и объём затрачиваемой оперативной памяти.

Реализовать программу на языке C или C++, соответствующую построенному алгоритму. Формат входных и выходных данных описан в варианте задания.

Вариант №1: Монеты. На первой строке заданы два числа, N и $p > 1$, определяющие набор монет некоторой страны с номиналами p_0, p_1, \dots, p_{N-1} . Нужно определить наименьшее количество монет, которое можно использовать для того, чтобы разменять заданную на второй строчке сумму денег M $2^{32} - 1$ и распечатать для каждого i -го номинала на i -ой строчке количество участвующих в размене монет. Кроме того, нужно обосновать почему жадный выбор неприменим в общем случае (когда номиналы могут быть любыми) и предложить алгоритм, работающий при любых входных данных.

1 Описание

Работа жадного алгоритма заключается в последовательном выборе локально оптимального решения, допуская, что конечное решение также окажется оптимальным.

Условия применения:

Общего критерия оценки применимости жадного алгоритма для решения конкретной задачи не существует, однако для задач, решаемых жадными алгоритмами, характерны две особенности: во-первых, к ним применим Принцип жадного выбора, а во-вторых, они обладают свойством Оптимальности для подзадач.

Говорят, что к оптимизационной задаче применим принцип **жадного выбора**, если последовательность локально оптимальных выборов даёт глобально оптимальное решение. В типичном случае доказательство оптимальности следует такой схеме:

- Доказывается, что жадный выбор на первом шаге не закрывает пути к оптимальному решению: для всякого решения есть другое, согласованное с жадным выбором и не хуже первого.
- Показывается, что подзадача, возникающая после жадного выбора на первом шаге, аналогична исходной.
- Рассуждение завершается по индукции.

Говорят, что задача обладает свойством **оптимальности для подзадач**, если оптимальное решение задачи содержит в себе оптимальные решения для всех её подзадач.

Решение. Для данной задачи жадный алгоритм не всегда даёт оптимальное решение, а только для некоторых, называемых каноническими, монетных систем, вроде используемых в США (1, 5, 10, 25 центов). Неканонические системы таким свойством не обладают. Так, например, сумму в 24 копейки монетами в 1, 5 и 7 коп. жадный алгоритм разменивает так: 7 коп. — 3 шт., 1 коп. — 3 шт., в то время как правильное решение — 7 коп. — 2 шт., 5 коп. — 2 шт. В нашем же случае, алгоритм всегда даёт оптимальное решение, так как номиналы состоят из степеней числа.

2 Исходный код

```
1 #include <iostream>
2 #include <cmath>
3 #include <vector>
4
5 int main(){
6     unsigned int n, m, num, first;
7     std::cin >> n;
8     if(n == 0){
9         return 0;
10    }
11    std::cin >> first;
12    std::cin >> m;
13
14    std::vector<int> p(n,0);
15    num = pow(first, n - 1);
16    int i = n - 1;
17    while(m > 0){
18        int count = 0;
19        while(m >= num){
20            ++count;
21            m -= num;
22        }
23        p[i] = count;
24        i -= 1;
25        num = pow(first,i);
26    }
27    for (auto j : p){
28        std::cout << j << '\n';
29    }
30    return 0;
31 }
```

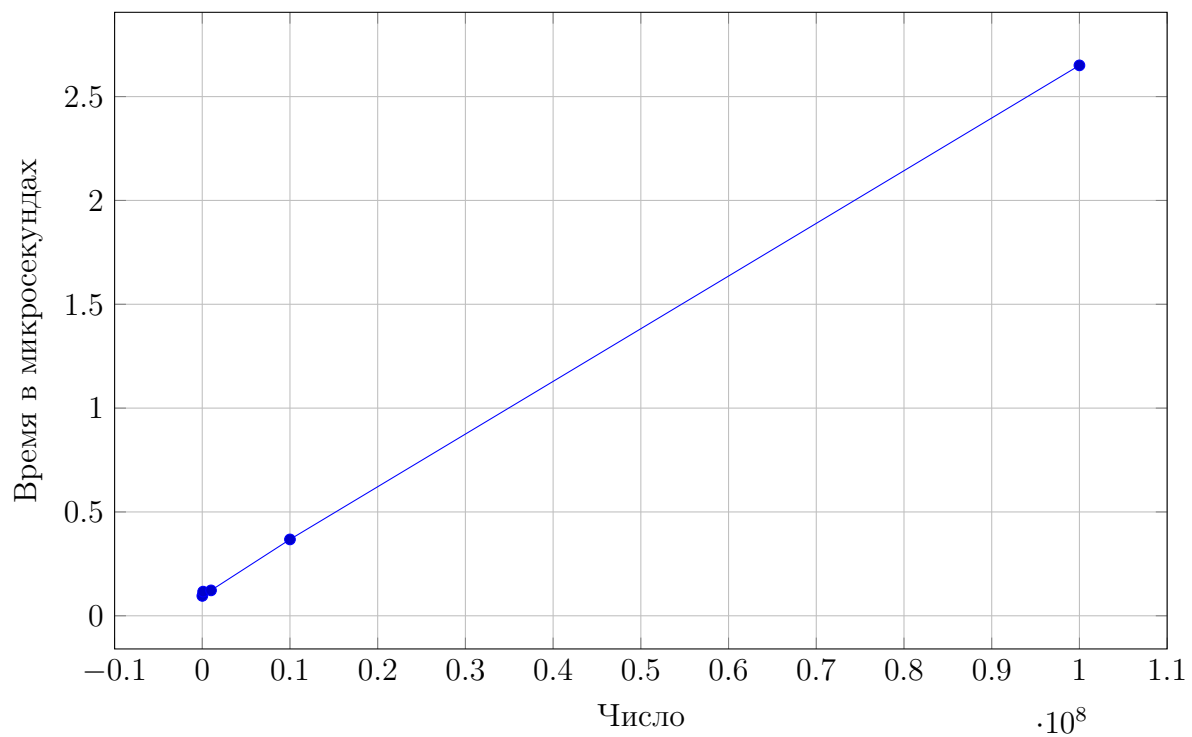
3 Тест производительности

Проверим сложность работы алгоритма. Алгоритм работает за линейное время $O(n)$, n - число, которое надо разменять.

1 Графики

Тестирование будет состоять из 5 тестов: где нужно будет разменять 1 1 10000/100000/1000000/10000000/100000000

Синий: line
·10⁴



2 Консольный вывод

test01.txt 10000

rusya@DESKTOP-93JGKCU:/mnt/c/Users/rusya/Desktop/lab8_DA./solution < test01.txt >
out.txt

Time: 863 ms

test02.txt 100000

rusya@DESKTOP-93JGKCU:/mnt/c/Users/rusya/Desktop/lab8_DA./solution < test02.txt >
out.txt

Time: 884 ms

test03.txt 1000000

rusya@DESKTOP-93JGKCU:/mnt/c/Users/rusya/Desktop/lab8_DA./solution < test03.txt >
out.txt

Time: 1284 ms

test04.txt 10000000

rusya@DESKTOP-93JGKCU:/mnt/c/Users/rusya/Desktop/lab8_DA./solution < test04.txt >
out.txt

Time: 3619 ms

test05.txt 100000000

rusya@DESKTOP-93JGKCU:/mnt/c/Users/rusya/Desktop/lab8_DA./solution < test05.txt >
out.txt

Time: 26540 ms

4 Выводы

В данной работе, я узнал о принципах работы жадных алгоритмов, условия, при которых они выполняются. Жадный алгоритм является довольно простым и эффективным решением, но для этого нужно доказать, что оптимальный выбор на каждом этапе даст нам оптимальное решение всей задаче.

Жадные алгоритмы ресурсоэкономный, в отличии от динамического программирования. Они широко используются в различных алгоритмах. Например, принцип жадного выбора используется в алгоритме Хаффмана(адаптивный алгоритм оптимального префиксного кодирования алфавита с минимальной избыточностью), алгоритме Крускала(поиск остовного дерева минимального веса в графе), алгоритме Прима(поиск остовного дерева минимального веса в связном графе).

Список литературы

- [1] Томас Х. Кормен, Чарльз И. Лейзерсон, Рональд Л. Ривест, Клиффорд Штайн. *Алгоритмы: построение и анализ, 2-е издание*. — Издательский дом «Вильямс», 2007. Перевод с английского: И. В. Красиков, Н. А. Орехова, В. Н. Романов. — 1296 с. (ISBN 5-8459-0857-4 (рус.))
- [2] Дональд Э. Кнут. *Искусство программирования. Том 3. Сортировка и поиск, 2-е издание*. — Правообладатель «Диалектика-Вильямс», 2018. Перевод с английского: И. В. Красиков, В. Т. Тертышный. — 834 с. (ISBN 978-5-8459-0082-1, 0-201-89685-0)
- [3] *Жадный алгоритм*
URL: https://ru.wikipedia.org/wiki/Жадный_алгоритм (дата обращения: 01.05.21).
- [4] *Chrono in C++*
URL: <https://www.geeksforgeeks.org/chrono-in-c/> (дата обращения: 01.05.21).