

Московский авиационный институт
(национальный исследовательский университет)

Факультет информационных технологий и прикладной
математики

Кафедра вычислительной математики и программирования

Лабораторная работа №8 по курсу «Дискретный анализ»

Студент: Т. А. Габдуллин
Преподаватель: А. А. Кухтичев
Группа: М8О-206Б
Дата:
Оценка:
Подпись:

Москва, 2019

Лабораторная работа №8

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.

Вариант: 3. Максимальный треугольник

Описание: Заданы длины N отрезков, необходимо выбрать три таких отрезка, которые образовывали бы треугольник с максимальной площадью.

Входные данные: В первой строке задано число n - количество отрезков. В следующих n строках записаны длины отрезков.

Выходные данные: Если никакого треугольника из заданных отрезков составить нельзя — 0, в противном случае на первой строке площадь треугольника с тремя знаками после запятой, на второй строке — длины трёх отрезков, составляющих этот треугольник. Длины должны быть отсортированы.

1 Метод решения

Итак, жадный алгоритм (greedy algorithm) — это алгоритм, который на каждом шагу делает локально наилучший выбор в надежде, что итоговое решение будет оптимальным. Алгоритмы, предназначенные для решения задач оптимизации, обычно представляют собой последовательность шагов, на каждом из которых предоставляется некоторое множество выборов. Определение наилучшего выбора, руководствуясь принципами динамического программирования, во многих задачах оптимизации напоминает стрельбу из пушки по воробьям; другими словами, для этих задач лучше подходят более простые и эффективные алгоритмы. В жадном алгоритме всегда делается выбор, который кажется самым лучшим в данный момент - т.е. производится локально оптимальный выбор в надежде, что он приведет к оптимальному решению глобальной задачи. Жадные алгоритмы не всегда приводят к оптимальному решению, но во многих задачах они дают нужный результат.

Описание взято с сайта *algorprog.ru*.

2 Реализация:

Очевидно, что элементарной подзадачей является *выбор стороны для треугольника*. Если имеется несколько доступных отрезков, то оптимальным выбором будет отрезок с наибольшей длиной, т.к. при этом мы получаем наибольший *полупериметр* для *Формулы Герона нахождения площади треугольника*. Поэтому, для решения задачи, отсортируем отрезки по длине и будем, начиная с самого длинного, брать их в качестве стороны, пока количество сторон не станет равно 3, проверяя условие треугольника $c < a + b$.

3 ЛИСТИНГ

```
1 | #include <bits/stdc++.h>
2 | using namespace std;
3 |
4 | double sqr(double a, double b, double c) {
5 |     double q = (a + b + c) / 2;
6 |     return sqrt(q * (q - a) * (q - b) * (q - c));
7 | }
8 |
9 | int main() {
10 |     int n;
11 |     cin >> n;
12 |     vector <int> a(n);
13 |     for(int i = 0; i < n; ++i)
14 |         cin >> a[i];
15 |     sort(a.rbegin(), a.rend());
16 |     double _max = 0;
17 |     int pos = 0;
18 |     for(int i = 0; i <= n - 3; ++i) {
19 |         if(a[i+1] + a[i+2] > a[i]) {
20 |             double current = sqr(a[i], a[i+1], a[i+2]);
21 |             if(current > _max) {
22 |                 pos = i;
23 |                 _max = current;
24 |             }
25 |         }
26 |     }
27 |     if(_max == 0) {
28 |         cout << 0 << endl;
29 |     } else {
30 |         cout << fixed << setprecision(3) << _max << endl;
31 |         for(int i = 2; i >= 0; --i) {
32 |             cout << a[pos+i] << " ";
33 |         }
34 |         cout << endl;
35 |     }
36 |     return 0;
37 | }
```

4 Выводы

Жадный алгоритм обладает достаточной мощностью и хорошо подходит для широкого класса задач, однако не всегда приводит к оптимальному решению задачи, но при этом к правильному.

Признаки того, что задачу возможно решить при помощи жадного алгоритма:

- Задачу можно разбить на подзадачи;
- Величины, рассматриваемые в задаче, можно дробить так же на подзадачи;
- Сумма оптимальных решений для двух подзадач даст оптимальное решения для всей задачи.

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] *Про жадные алгоритмы*
URL:<https://algotprog.ru/material/module-25226>