**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №1**

**по дисциплине «Построение и анализ алгоритмов»**

Тема: «Поиск с возвратом»

|  |  |  |
| --- | --- | --- |
| Студент гр. 6381 |  | Попов Н.В. |
| Преподаватель |  | Филатов А.Ю. |

Санкт-Петербург

2018

**Цель работы.**

Изучение алгоритма поиска на основе задачи минимального квадрирования квадрата размером NxN.

**Описание алгоритма.**

На вход программа получает число – размер квадрата, который нужно разделить на несколько меньших квадратов. После этого совершается проверка на делимость этого числа на 2, 3 и 5 и квадрирование заданного квадрата на 4, 6 и 8 частей соответственно. Если ни одно из этих чисел не является делителем размера квадрата, то для сокращения объема вычислений часть заданной фигуры заполняется квадратами размерами (N/2)-1, двумя (N/2) и одним небольшим размером 1, примыкающим к первому и одному из вторых. Далее оставшийся пустой участок заполняется алгоритмом поиска с возвратом, т. е. в незанятую клетку вставляются квадраты разных размеров и результаты этих вставок сравниваются между собой. Результат, содержащий наименьшее количество квадратов, выводится в форме списка этих самых квадратов, содержащий их координаты и размеры.

**Описание функций.**

* void Board::printBestBrd() – вывод результата с наименьшим количеством квадратов.
* bool Board::findFreeCell(int &x, int &y) – поиск свободной клетки для вставки квадрата. При положительном результате возвращает 1.
* bool Board::exitFromBounds(int x, int y, int width) – проверка на выход квадрата размером width из границ доски. Возвращает 1 при положительном результате.
* bool Board::buildSq(int x, int y, int width) – проверка на возможность построения квадрата, и при таковой возможности строит его.
* void Board::deleteSq(int x, int y) – удаление квадрата из массива busyCells класса Board, т. е. занесение в массив нулей.
* void Board::changeBestBoard() – смена набора наименьшего количества квадратов bestBoard на текущий busyCells.
* void squaring(Board \*board, int x = 0, int y = 0) – разделение квадрата на минимальное количество квадратов меньших размеров.
* void modSquaring(Board \*board) – сравнение размера заданного квадрата по модулю 2, 3 и 5. При результате равным нулю вывод результата квадрирования, иначе вставка 4ех первоначальных квадратов и дальнейшее квадрирование вручную функцией squaring.
* int Board::findSq(int& x, int& y, int num) – вспомогательная функция для ф-ии printList. Находит координаты квадрата с номером num.
* void printList(Board \*board) – вывод набора наименьшего количества квадратов bestBoard списком, содержащим координаты и размер.

**Тестирование.**

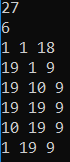
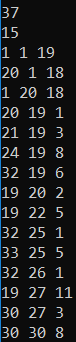
   

Рисунок 1. Результаты для 7, 16, 27 и 37 размерных квадратов соот.

**Вывод.**

В процессе выполнения данной лабораторной работы был изучен алгоритм поиска с возвратом и с помощью него была решена задача минимального квадрирования квадрата размером NxN.