Государственное образовательное учреждение высшего профессионального образования
«Московский государственный технический университет имени Н.Э. Баумана»
(МГТУ им. Н.Э. Баумана)

Рубежный контроль №1

по курсу «Анализ алгоритмов»

Ковер Серпинского

Выполнил: Тимонин А.С., гр. ИУ7-52Б

Преподаватели: Волкова Л.Л., Строганов Ю.В.

2019 г.

Оглавление

Введение

Фракталы вокруг нас повсюду, и в очертаниях гор, и в извилистой линии морского берега. Некоторые из фракталов непрерывно меняются, подобно движущимся облакам или мерцающему пламени, в то время как другие, подобно деревьям или нашим сосудистым системам, сохраняют структуру, приобретенную в процессе эволюции. Х. О. Пайген и П. Х. Рихтер.

Задача рубежного контроля:

- Разработать и реализовать программу, которая строит фрактал под названием ковер Серпинского.
- Протестировать полученное программное обеспечение.
- Исследовать полученное рпограммное обеспечение.

1. Аналитический раздел

В данном разделе будет дано определение фрактала ковра Серпинского, а также будет описано его построение. В данном разделе будуд описаны алгоритмы Кнута-Морриса-Пратта и Байера-Мура.

1.1 Ковер Серпинского

Ковёр Серпинского (квадрат Серпинского) — фрактал, один из двумерных аналогов множества Кантора, предложенный польским математиком Вацлавом Серпинским.

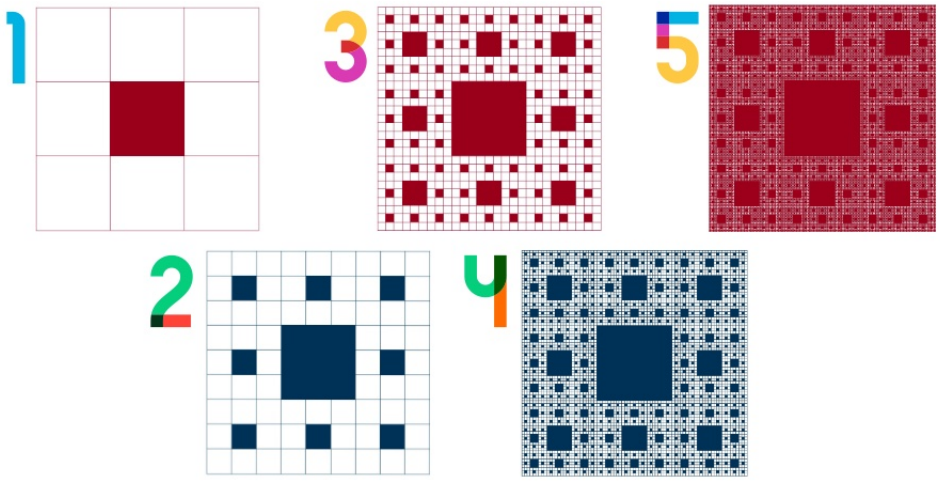

Рис. 1.1: Построенный ковер Серпинского со степенями фрактал от 1 до 5

1.2 Как строится ковер Серпинского

Вычислим площадь ковра Серпинского, считая исходный квадрат единичным. Для этого достаточно вычислить площадь вырезаемых квадратов. На первом шаге вырезается квадрат площади 1/9. На втором шаге вырезается восемь квадратов, каждый из которых имеет площадь 1/81. На каждом следующем шаге число вырезаемых квадратов увеличивается в восемь раз, а площадь каждого из них уменьшается в девять раз. Таким образом, общая площадь вырезаемых квадратов представляет собой сумму геометрической прогрессий с начальным членом 1/9 и знаменателем 8/9. По формуле суммы геометрической прогрессии находим, что это число равно единице, т. е. площадь ковра Серпинского равна нулю.

1.3 Вывод

Было дано определение фракталу Серпинского, также было описано как данный фрактал строится.

2. Технологический раздел

В данном разделе будут рассмотренны требования к разрабатываемому программному обеспечению, средства, использованные в процессе разработки для реализации поставленных задач, а также представлены листинги кода программы.

2.1 Требования к программному обеспечению

Программное обеспечение должно реализовывать алгоритм построения фрактала ковра Серпинского в двух реализациях: рекурсивная и итеративная.

2.2 Средства реализации

Для выполнения поставленной задачи был использован язык программирования C++. Среда для разработки QtCreator.

Данная среда разработки связана с тем, что в ней можно удобно реализовывать отрисовку пикселей и выводить картинку не в консоль, а в собственое окно.

Версия компилятора C++: GNU++14 [-std=gnu++14]

2.3 Листинг кода

В данном подразделе приведены листинги кода построения фрактала Серпинского в рекурсивной и итеративной реализациях. 2.1-2.2 реализации алгоритмов.

Листинг 2.1: Рекурсивное построение ковра Серпинского

```
1  void paintwidget::kov(QPainter &painter, double A, double B, double C, double D, int n)
2  {
3  if (delay) {
4  QTime end = QTime::currentTime().addMSecs(1);
5  while(QTime::currentTime() < end)
6  QCoreApplication::processEvents(QEventLoop::AllEvents, 1000);
7  repaint();
8  }
9
10 draw_rect(painter, A, B, C, D);
11
12 double A1, B1, C1, D1;
13 if(n > 0) {
14 A1 = 2 * A / 3 + C / 3;
15 C1 = A / 3 + 2 * C / 3;
16 B1 = 2 * B / 3 + D / 3;
17 D1 = B / 3 + 2 * D / 3;
```

```
18
19 int x1 = A1, y1 = B1, x2 = C1, y2 = D1;
20 //painter.drawRect(A1, B1, C1, D1);
21
22
23 int miny = std::min(y1, y2);
24 int maxy = std::max(y1, y2);
25
26 for (int i = miny; i < maxy; i++) {
27 painter.drawLine(x1, i, x2, i);
28 }
29
30 kov(painter, A, B, A1, B1, n−1);
31 kov(painter, A1, B, C1, B1, n−1);
32 kov(painter, C1, B, C, B1, n−1);
33 kov(painter, A, B1, A1, D1, n−1);
34 kov(painter, C1, B1, C, D1, n−1);
35 kov(painter, A, D1, A1, D, n−1);
36 kov(painter, A1, D1, C1, D, n−1);
37 kov(painter, C1, D1, C, D, n−1);
38 }
39 }
```

Листинг 2.2: Итеративное построение ковра Серпинского

```
1 void paintwidget::nkov(QPainter &painter, double A, double B, double C, double D, int n)
2 {
3 double A1, C1, B1, D1;
4 QStack<fract_type> stck;
5
6 stck.push({A, B, C, D, n});
7
8 while (!stck.isEmpty()) {
9 fract_type tmp = stck.pop();
10 if (delay) {
11 QTime end = QTime::currentTime().addMSecs(1);
12 while(QTime::currentTime() < end)
13 QCoreApplication::processEvents(QEventLoop::AllEvents, 1000);
14 repaint();
15 }
16
17 draw_rect(painter, tmp.A, tmp.B, tmp.C, tmp.D);\
18 if (tmp.n > 0) {
19 A1 = 2 * tmp.A / 3 + tmp.C / 3;
20 C1 = tmp.A / 3 + 2 * tmp.C / 3;
21 B1 = 2 * tmp.B / 3 + tmp.D / 3;
22 D1 = tmp.B / 3 + 2 * tmp.D / 3;
23
24 int x1 = A1, y1 = B1, x2 = C1, y2 = D1;
25
26 int miny = std::min(y1, y2);
27 int maxy = std::max(y1, y2);
28
29 for (int i = miny; i < maxy; i++) {
```

```
30 painter.drawLine(x1, i, x2, i);
31 }
32
33 stck.push({C1, D1, tmp.C, tmp.D, tmp.n − 1});
34 stck.push({A1, D1, C1, tmp.D, tmp.n − 1});
35 stck.push({tmp.A, D1, A1, tmp.D, tmp.n − 1});
36 stck.push({C1, B1, tmp.C, D1, tmp.n − 1});
37 stck.push({tmp.A, B1, A1, D1, tmp.n − 1});
38 stck.push({C1, tmp.B, tmp.C, B1, tmp.n − 1});
39 stck.push({A1, tmp.B, C1, B1, tmp.n − 1});
40 stck.push({tmp.A, tmp.B, A1, B1, tmp.n − 1});
41 }
42 }
43 }
```

3. Экспериментальный раздел

В экспериментальном разделе будут приведен график сравнения рекурсивной и итеративной реализации построения фрактала Серпинского.

3.1 Сравнительный анализ

На Рис. 3.1 приведен график зависимости процессорного времени от степени фрактала для двух различных реализаций построения фрактала.

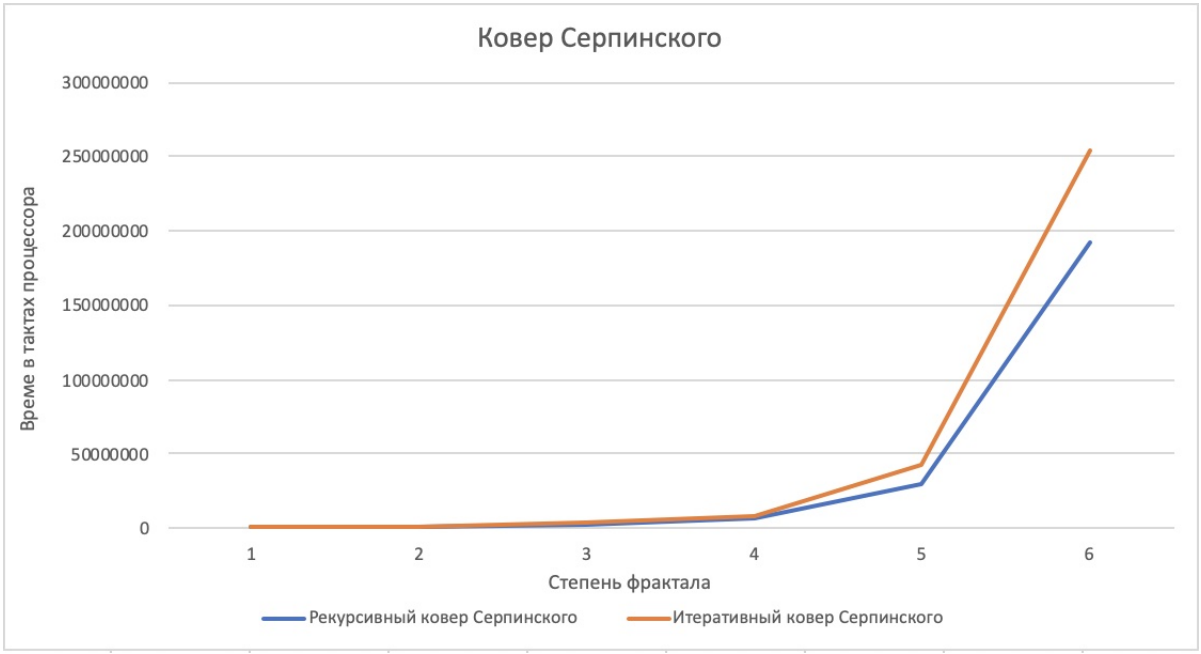

Рис. 3.1: График зависимости процессорного времени от степени фрактала

3.2 Вывод

Был приведен график зависимсти процессорного времени от степени фрактала для двух реализаций: рекурсивной и итеративной.

Приложения

В данном разделе будут приведены рисунки 3.2-3.2 с фракталами в разных степенях рекурсии.

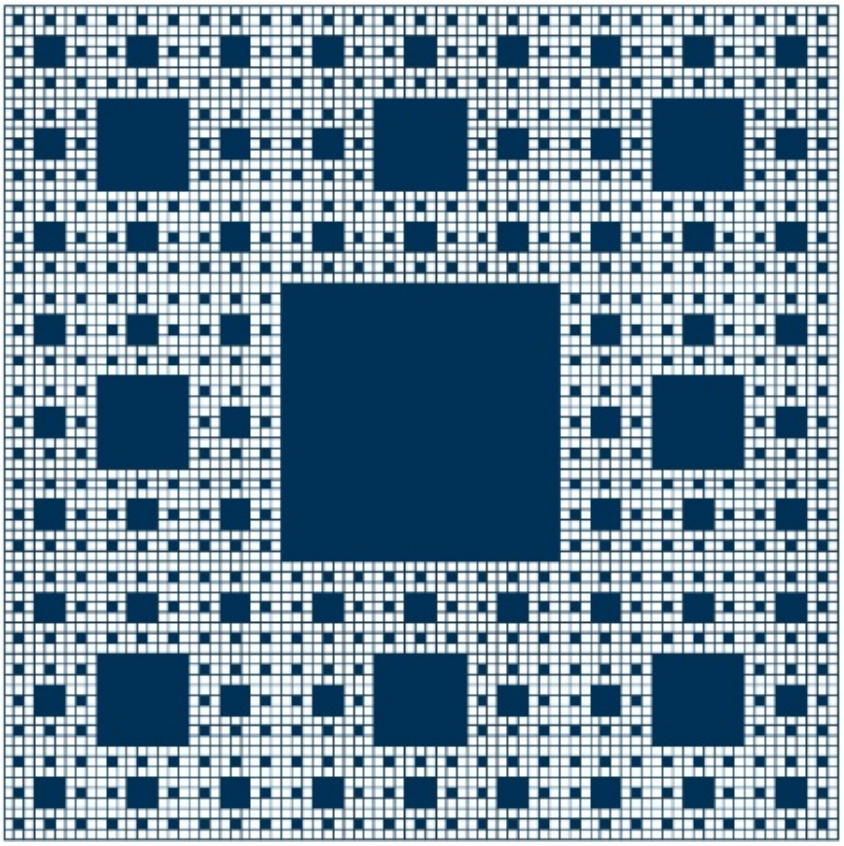

Рис. 3.2: Степень 4 фрактала ковра Серпинского

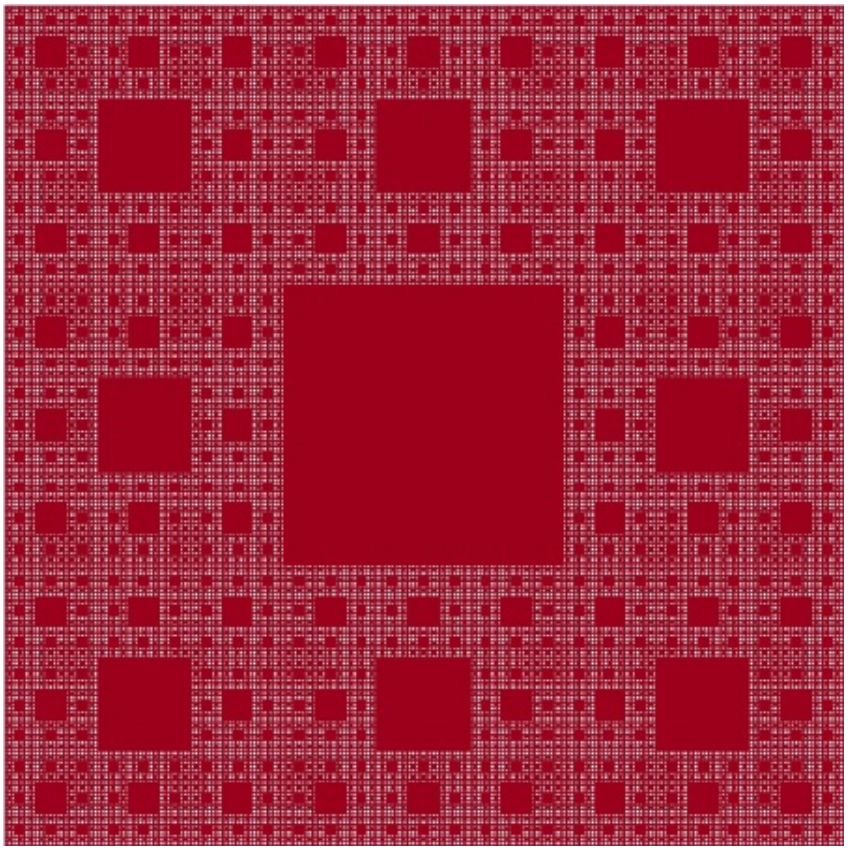

Рис. 3.3: Степень 5 фрактала ковра Серпинского