



Informatics II

Exercise 2

Feb 24, 2020

Recursion

Task 1. Consider a finite sequence $s(n) = a_1, a_2, \dots, a_m$ with $a_1 = n$ and $a_m = 1$ that is defined as follows:

$$a_{i+1} = \begin{cases} a_i/2 & \text{if } a_i \text{ is even} \\ 3a_i + 1 & \text{if } a_i \text{ is odd} \end{cases}$$

1. Determine $s(3)$.
2. Write a C program with a recursive function **void sequence(int n)** that computes and prints all elements of $s(n)$.

Task 2. Let B_k be the set of binary strings of length k that do not include consecutive 1's.

1. Determine B_3 .
2. Write a C program with a recursive function that reads length k from the command line and prints B_k .

Task 3. A T-square fractal is a two-dimensional fractal as illustrated in Figure 1. The figure shows the first four iterations of the T-square fractal.

- (a) Implement a recursive C function **void TSquareFractal(double x, double y, double l, int i)**, where (x, y) is the coordinate of the lower left corner and l is the side length of a square. As output the function **TSquareFractal** shall print out the coordinates of all squares together with the length of the side.

Example: The invocation **TSquareFractal(0, 0, 100, 2)** prints the coordinates of five squares and their side length drawn till iteration 2 as depicted in 1b. The output would be:

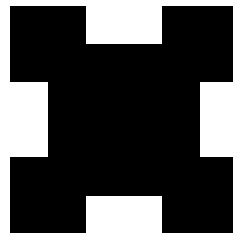
```
(0.00, 0.00), 100.00
(-25.00, -25.00), 50.00
(-25.00, 75.00), 50.00
(75.00, -25.00), 50.00
(75.00, 75.00), 50.00
```

- (b) Write a C program that uses function **void TSquareFractal(double x, double y, double l, int i)** to draw the T-Square Fractal using SDL library.

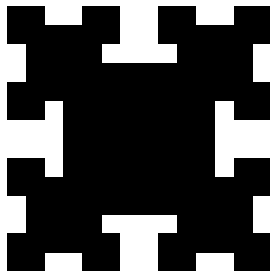
Note: A skeleton of the solution for Task 3(b) is provided.



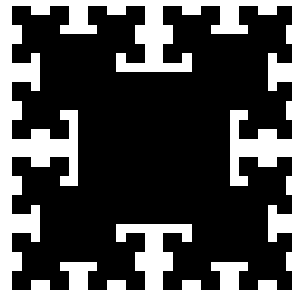
(a) Iteration =1



(b) Iteration 2



(c) Iteration 3



(d) Iteration 4

Figure 1: T-square fractal of first four iterations