CS 101 Computer Programming and Utilization

Practice Problems

Param Rathour
https://paramrathour.github.io/CS101

Autumn Semester 2020-21

Last update: 2020-12-26 16:50:56+05:30

Disclaimer

These are **optional** problems. As these problems are pretty involving, my advice to you would be to first solve exercises given in slides and get comfortable with the course content. The taught methods will suffice to solve these problems. (You are free to use 'other' stuff but not recommended)

Contents

§1. Practice Problems 1

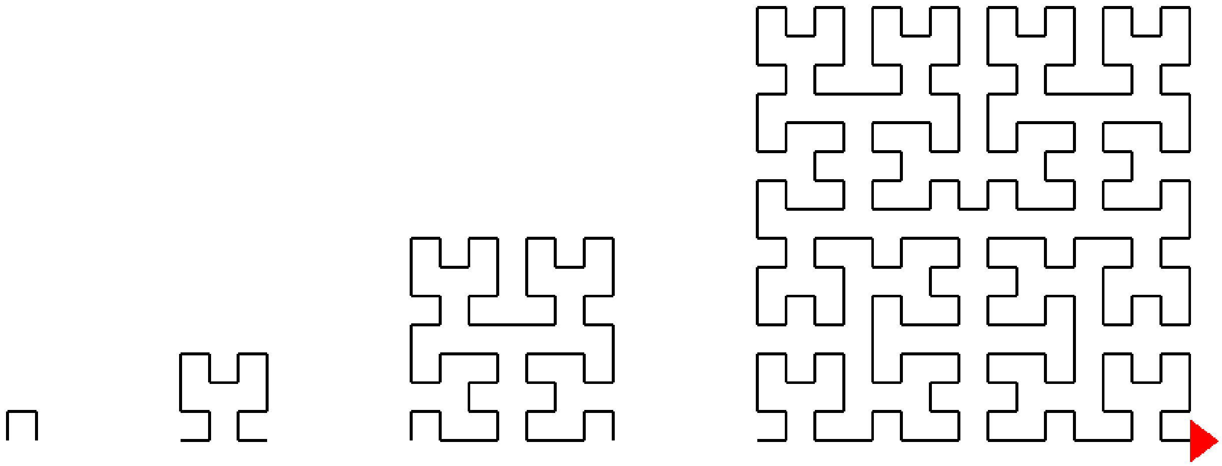

Figure 1: Hilbert Curve

1. **Problem Statement:**
   Take an integer as input and draw the corresponding iteration of this fractal using turtleSim
   You may think along these lines

   Step 1 Find a simple pattern in these iterations

   Step 2 Think how can you implement this pattern in an efficient way (here think in the number of lines of code you have to write. **Word of caution**: this is just one of the possible definitions of efficient code)

   Step 3 Do you think that you need something that will implement/shorten your code?
   How will it look like? (it's a feature)

   Feel free to discuss your thoughts on this.

   Note. *For people comfortable with the basics of C++, this shouldn't be difficult. You may try this*

Fun Videos

Hilbert's Curve: Is infinite math useful?
Recursive PowerPoint Presentations [Gone Fractal!]

Book Chapters for Graphics

Additional chapters of the book on Simplecpp graphics demonstrating its power
(It is just a list, you are not expected to understand/study things, CS101 is for a reason :P)

Chapter 1 Turtle graphics

Chapter 5 Coordinate based graphics, shapes besides turtles

Chapter 15.2.3 Polygons

Chapter 19 Gravitational simulation

Chapter 20 Events, Frames, Snake game

Chapter 24.2 Layout of math formulae

Chapter 26 Composite class

Chapter 28 Airport simulation

§2. Practice Problems 2

1. You probably heard about Fibonacci Numbers!

   The Fibonacci numbers are the numbers in the integer sequence: (defined by the recurrence relation)

$$
\begin{aligned}
&F_0 = 0 \\
&F_1 = 1 \\
&F_n = F_{n-1} + F_{n-2} \quad n \in \mathbb{Z} \quad \text{(They can be extended to negative numbers)}
\end{aligned} \tag{1}
$$

   For any integer n, the sequence of Fibonacci numbers F_i taken modulo n is periodic.

   The Pisano period, denoted $\pi(n)$, is the length of the period of this sequence.

   For example, the sequence of Fibonacci numbers modulo 3 begins:

$$0, 1, 1, 2, 0, 2, 2, 1, 0, 1, 1, 2, 0, 2, 2, 1, 0, 1, 1, 2, 0, 2, 2, 1, 0, \ldots \text{(A082115)}$$

   This sequence has period 8, so $\pi(3) = 8$. (Basically, the remainder when these numbers are divided by n is a repeating sequence. You have to find the length of sequence)

   Problem Statement:

   (a) Find Pisano period of n numbers $k_1, k_2, \ldots, k_n$

   Input Format
   n
   $k_1, k_2, \ldots, k_n$
   Output Format
   $\pi(k_i)$ (each on a newline)
   Sample Input
   3
   3 10 25
   Sample Output
   8
   60
   100

   (b) For n numbers $k_1, k_2, \ldots, k_n$, find $\max(\pi(i))$ for $i = 1, 2, \ldots, k$ and corresponding i
   If there are 2 (or more) such i's, output smallest of them

   Input Format
   n
   $k_1, k_2, \ldots, k_n$
   Output Format
   k $\pi(k)$ (each pair on a newline)
   Here k is smallest possible integer satisfying $\pi(k) = \max(\pi(i))$ for possible i
   Sample Input
   5
   20 40 60 80 100
   Sample Output
   10 60
   30 120
   50 300
   50 300
   98 336

2. $$\frac{\pi}{2} = \sum_{k=0}^{\infty} \frac{k!}{(2k+1)!!} = \sum_{k=0}^{\infty} \frac{2^k k!^2}{(2k+1)!} \tag{2}$$

Note. *$n!!$ is called double factorial. $n!! \neq (n!)!$.*

Problem Statement:
Calculate π using first $k_i + 1$ terms [1] of Equation (2) for n different natural numbers $k_1, k_2, \ldots, k_n$.
Give your answers correct to 10 decimal places

Input Format
n
$k_1, k_2, \ldots, k_n$
Output Format
Calculated π for k_i (each on a newline)
Sample Input
3
10 20 30
Sample Output
3.1411060216
3.1415922987
3.1415926533

3. **Simpson's Rule**: a method for numerical integration

$$\int_a^b f(x)\,dx \approx \frac{\Delta x}{3}\left(f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + 2f(x_4) + \cdots + 4f(x_{n-1}) + f(x_n)\right) \tag{3}$$

Note. *Simpson's rule can only be applied when an odd number of ordinates is chosen.*

Problem Statement:
Solve Equation (4) giving the answers correct to 7 decimal places (Use 101 ordinates)

$$\int_{0.5}^{1} \frac{\sin\theta}{\theta}\,\mathrm{d}\theta \tag{4}$$

Correct Answer = 0.4529756

[1] i.e calculate π till $\dfrac{k_i!}{(2k_i+1)!!}$ term

§3. Practice Problems 3

1. Write a function that calculates the day of the week for any particular date in the past or future. Consider Gregorian calendar (AD)

   Task 1: As a programming exercise, try the naive approach:
   Starting from 1 Jan 0001 (Saturday) and calculate day after day till you reach the given date
   Use `switch-case` statement

   Task 2: Try to make more efficent algorithm (reduce completion time) than Task 1
   Implement it, and discuss your approach with me.

   Also check for invalid dates (Write another function for this)
   If dates are invalid, output **-1**

   Input Format
   n
   Followed by n dates in **Date Month Year** format
   Output Format
   Day of the Week
   Sample Input
   5
   19 2 1627
   29 2 1700
   15 4 1707
   22 12 1887
   23 6 1912
   Sample Output
   Monday
   -1
   Friday
   Thursday
   Sunday

   Note. *Use your Task 1 program to check Task 2 implementation.*

2. **Farey Sequence**
   This sequence has all rational numbers in range $[0/1$ to $1/1]$ sorted *in increasing order* such that the denominators are less than or equal to n and all numbers are in *reduced forms* i.e., 2/4 does not belong to this sequence as it can be reduced to 1/2.

   Input Format
   n
   Output Format
   Corresponding numbers in sequence in p/q format
   Sample Input
   7
   Sample Output
   0/1 1/7 1/6 1/5 1/4 2/7 1/3 2/5 3/7 1/2 4/7 3/5 2/3 5/7 3/4 4/5 5/6 6/7 1/1

   Can you find efficient solution?

Fun Video

Funny Fractions and Ford Circles

3. **Thue-Morse Sequence aka Fair Share Sequence**
   Thue-Morse Sequence is the infinite binary sequence obtained by starting with 0 and successively appending the Boolean complement of the sequence obtained thus far (called prefixes of the sequence).
   First few steps :

   - Start with 0
   - Append complement of 0, we get 01
   - Append complement of 01, we get 0110
   - Append complement of 0110, we get 01101001

   Problem Statement:
   Consider appending complement of a prefix to itself as one iteration
   Define a function to take a positive integer n as input then iterate n times to print the first 2^n digits

Input Format
n
Output Format
Corresponding digits in sequence
Sample Input
6
Sample Output
0110100110010110100101100110100110010110011010010110100110010110

   Again, can you find better solution?

   ## Fun Video

   The Fairest Sharing Sequence Ever

4. **Collatz Conjecture**
   Consider the following operation on an arbitrary positive integer:

   - If the number is even, divide it by two.
   - If the number is odd, triple it and add one.

   Collatz Conjecture states that no matter which positive integer we start with; we always end up with 1.

   Problem Statement:
   Define a function which performs this operation repeatedly on the result at each step; beginning with a given input n ($n < 10^6$), returns the number of operations required to reach 1[2]

Input Format
Arbitrary number of testcases (each space separated)
Stop when input is negative
Output Format
Count of operations for each number (each on a newline)
Sample Input
1 3 7 9 27 871 77031 -1
Sample Output
0
7
16
19
111
178
350

[2]As of 2020, the conjecture has been checked by computer for all starting values up to $2^{68} \approx 2.95 \times 10^{20}$, so sequence from n will reach 1

§4. Hints

§§4.1. Practice Problems 1

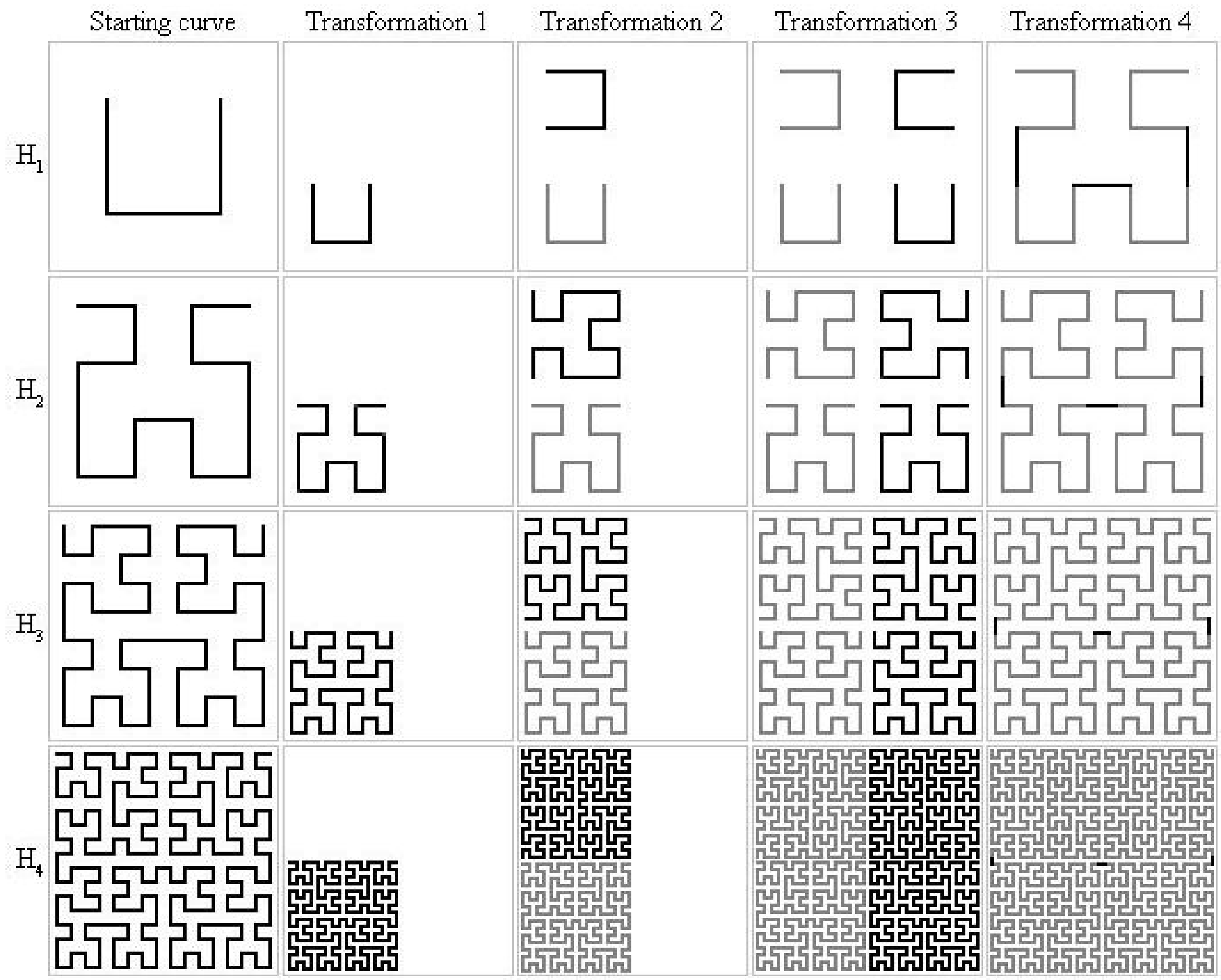

Figure 2: Hilbert Curve Transformations

§§4.2. Practice Problems 2

1. The sequence repeats when we encounter '0' then '1'
2. Similar to calculating e (problem given in slides)

§§4.3. Practice Problems 3

1. Try skipping days, months, years and centuries :D

   Is there a formula?
2. There is a pattern between 3 consecutive terms
3. How will you calculate n^{th} term?

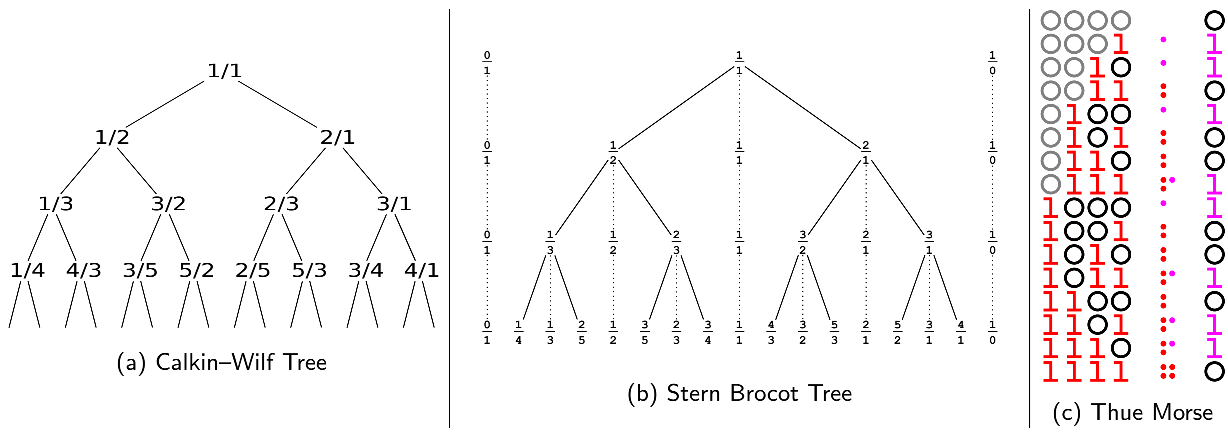
1/1
1/2
2/1
1/3
3/2
2/3
3/1
1/4
4/3
3/5
5/2
2/5
5/3
3/4
4/1
(a) Calkin–Wilf Tree
(b) Stern Brocot Tree
(c) Thue Morse

Figure 3