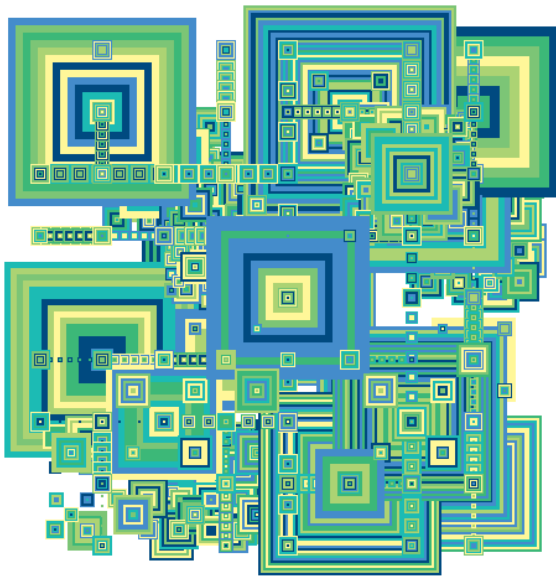


# Homework #02: Computational Arts

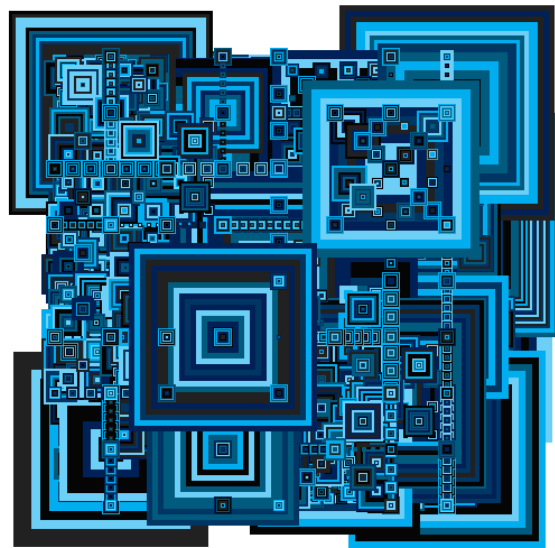
Computer Graphics  
School of Software  
College of Computer Science, Kookmin University

## Abstract

In this assignment, we will create a program that generates *Abstract Geometric Art*, as depicted in Fig. 1, with Modern OpenGL programming.



(a) 15\_03 11/2/2007 by Don Relyea



(b) Blue Recursive 11/1/2007 for www.nanohedron.com

Fig. 1. Space Filling Curve Art by Don Relyea (see details in Don Relyea's homepage)

## 1 INTRODUCTION

The goal of this homework is to write your own program that generates a set of rectangles in an artistic way.

When you carefully observe Fig. 1, you may realize the following hidden rules;

- 1) It consists of several groups of squares. Each group consists of a number of squares, in a recursive way. In the recursion, a larger square contains a smaller square while the distance between the larger square and the smaller square is a fixed one.
- 2) The placements of square groups do not look like a simple random. There seems a rule about the 2D placements of square groups.

The first rule is called 'offsetting' (see Fig. 2(a)), and the second rule is related with 'Hilbert space filling curve' (see Fig. 2(b)).

## 2 REQUIREMENTS

Your program should have the following subroutines. It is also ok for you to define your own subroutine additionally.

- 1) Your program contains a subroutine that select a set of colors for each square group
- 2) Your program contains a subroutine that generates a group of squares

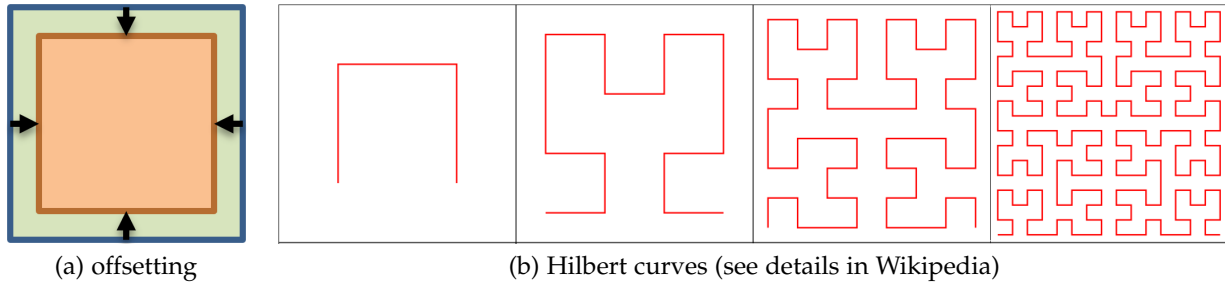


Fig. 2. Some concepts helpful to accomplish your homework

- Input
    - 2D Placement of Group (i.e., group center)
    - A number of squares in a group
    - A offset value
    - A set of colors
  - Output
    - A group of squares
- 3) Your program contains a subroutine that generates the Hilbert curves as shown in Fig. 2(b) in a hierarchical way.
- Output
    - Randomly selected 2D points on the curve in each level of Hilbert curves.
      - \* For a course level of Hilbert curves
        - It is better to select a small number of random 2D positions.
        - For each selected 2D position, it is better to generate a square group in a way that the size of the largest square is properly big enough.
      - \* For a fine level of Hilbert curve
        - It is better to select a large number of random 2D positions.
        - For each selected 2D position, it is better to generate a square group in a way that the size of the largest square is properly small enough.
- 4) In your program, you can change several parameters by using the context menu, in a way that we select one of the pre-defined parameters in the context menu. For example,
- Offsetting
    - 7
    - 5
    - 3
  - Min/Max numbers of square groups in each level
    - 1 / 5
    - 5 / 15
    - 15 / 30
  - Ratio of square group numbers between consecutive levels
    - 2
    - 3
    - 5

### 3 WHAT YOU LEARN

From this homework you can learn about the followings

- How to visualize a scene with Modern OpenGL
  - Write a your own shader program consisting of a vertex shader and a fragment shader
  - Use Modern OpenGL functions, related to drawing polygons with shaders, such as

- \* glUseProgram
- \* glUniform
- \* glEnableVertexAttribArray, glDisableVertexAttribArray
- \* glVertexAttribPointer,
- \* glDrawArrays

- How to transform an object in the scene with a model matrix, by using your transformation functions.
- How to manipulate a your virtual camera with a view matrix, by using your own lookAt() function.
- How to make a menu with FreeGLUT.
  - Use glutCreateMenu, glutAddMenuEntry, glutAttachMenu

#### 4 DEADLINE AND MISC. (EXTREMELY IMPORTANT!!!)

- This assignment can be accomplished in a team, typically composed with 1 or 2 people.
  - Caution 1: Once you have a team, you cannot change it until the end of the semester.
  - Caution 2: Only one of your team members submits your homework.
- Your homework must be submitted to 가상대학, until 23:59 on May. 1.
- You should compress the followings into a **tar.gz** or **zip** file whose name is OOOOOOOO\_HW02.tar.gz (OOOOOOOO should be your student ID).
  - 1) Your source files
  - 2) Makefile
  - 3) README.txt file, including the student numbers of the team members.
- **Your source codes must be complied with a make command in Ubuntu 16.04 LTS.**
- If you ask some questions to me, please utilize the office hours. It is also fine that you contact T.A. for asking some questions about the homework.