Advanced Programming

Programming Assignment #3

Kang Hoon Lee

Kwangwoon University

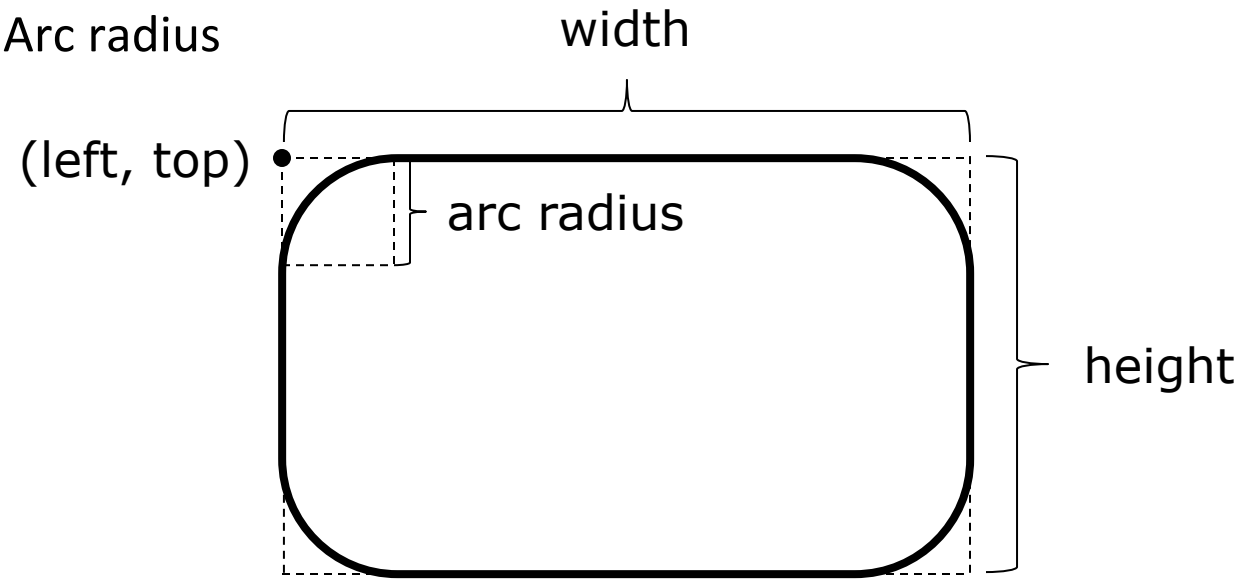
Basic Requirements

- **Add 3 new shape classes in `Graph.h` and `Graph.cpp`**
  - **`Box`**
  - **`Regular_polygon`**
  - **`Star`**

- **Define each new class by the following steps**
  - Derive from the **`Shape`** class
  - Add member variables (**`private`**)
  - Add getter functions (**`public`**)
  - **No** additional setter functions!
  - Define a constructor
  - Override **`draw_lines`**

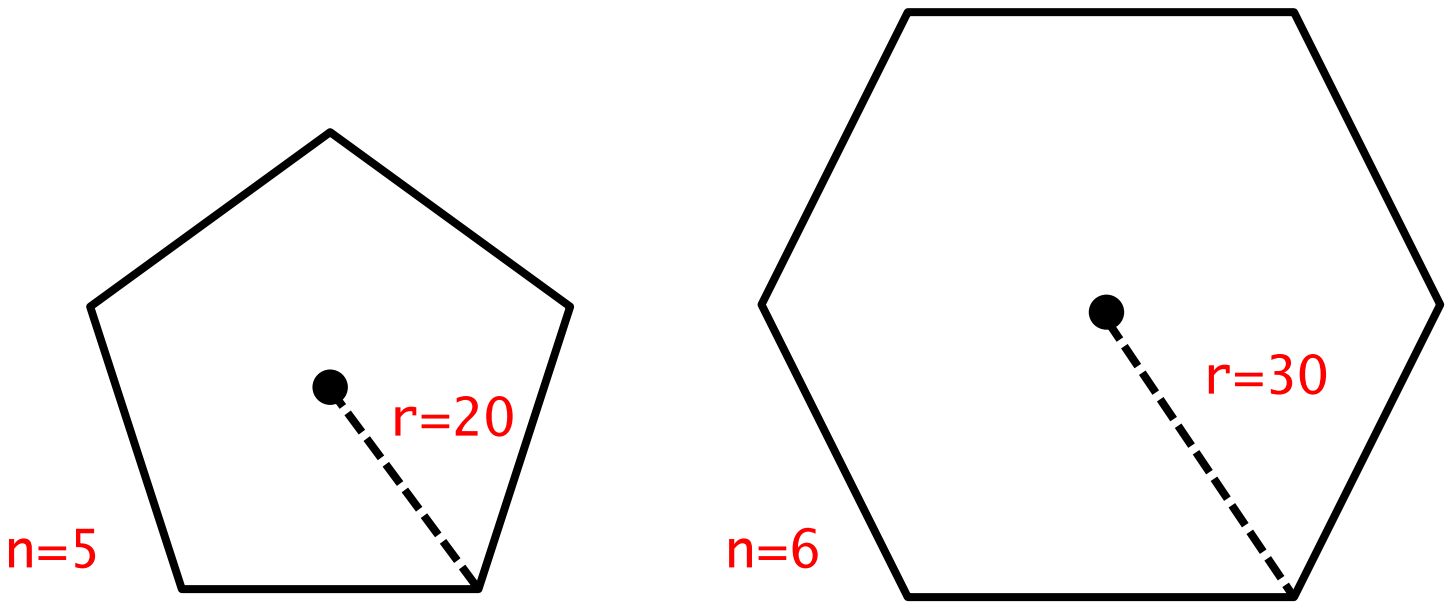
Basic Requirements

- **Box**
  - Left-top position (stored in **`Shape::points`**)
  - Width
  - Height
  - Arc radius



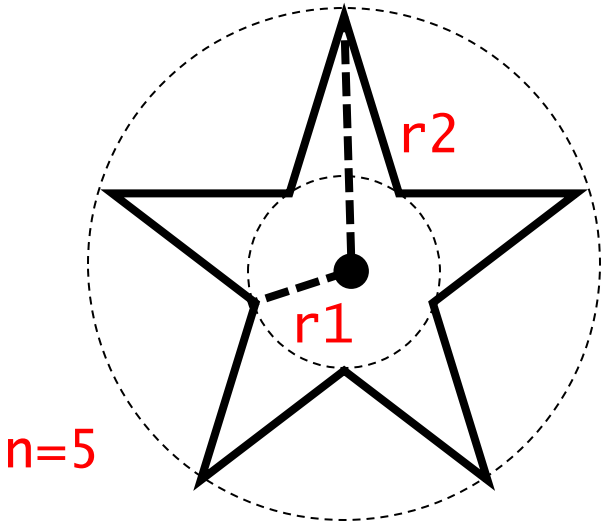
Basic Requirements

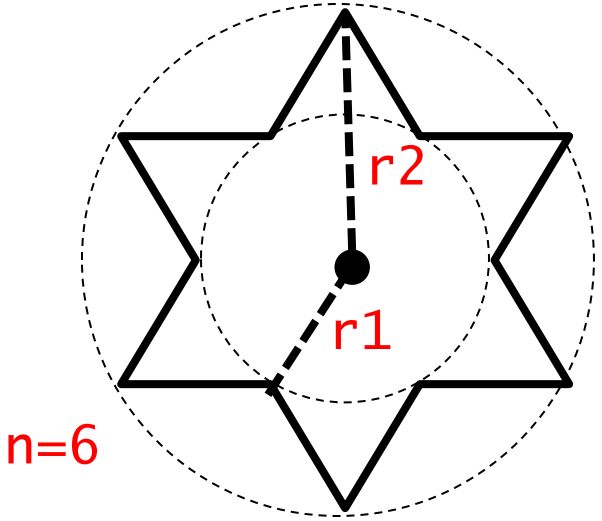
- **`Regular_polygon`**
  - Center position (stored in **`Shape::points`**)
  - Number of sides (**`n`** > 2, **`int`**)
  - Distance from the center to the corner (**`r`**, **`double`**)



Basic Requirements

- **`Star`**
  - Center position (stored in **`Shape::points`**)
  - Number of vertices (n, >2)
  - Distance from the center to the inner corner (r1)
  - Distance from the center to the outer corner (r2)



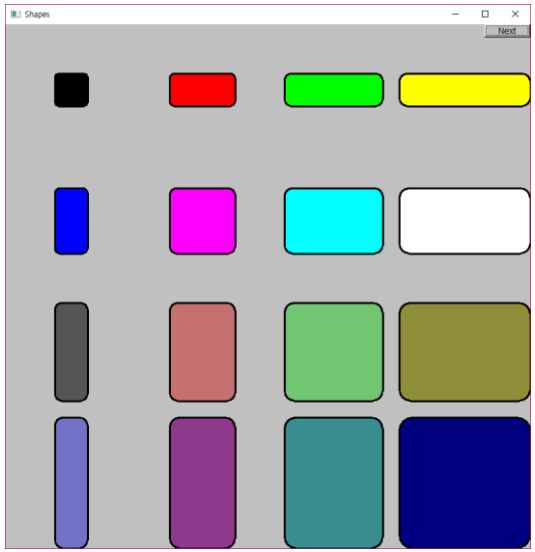


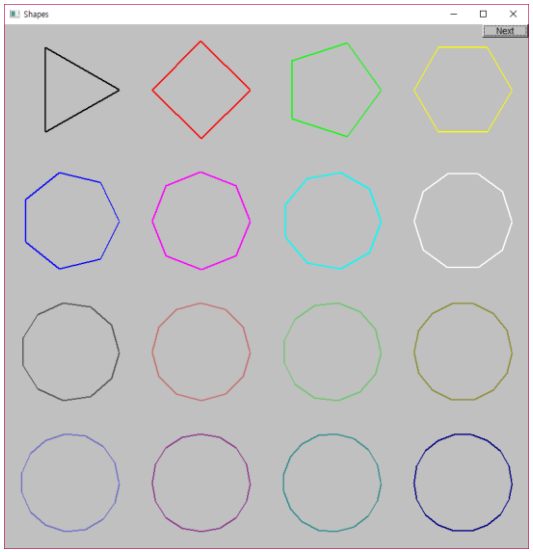
Basic Requirements

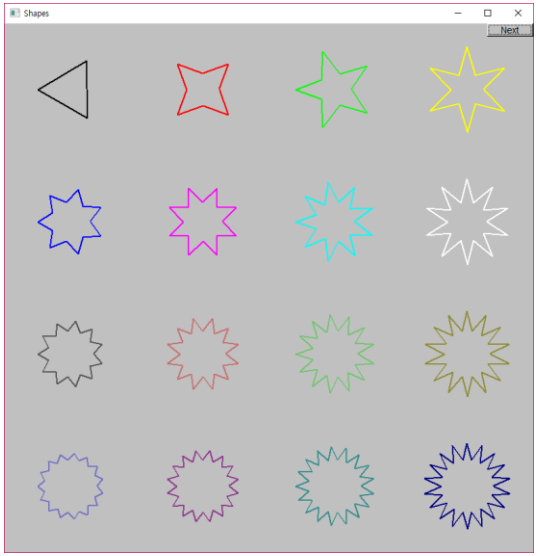
- **Write a simple test program `main_shapes.cpp`**
  - Create a window (resolution: 800x800)
  - Demonstrate the **`Box`** class
    - Attach more than 4x4 boxes of different properties
    - Wait for next button
    - Detach all of the boxes
  - Demonstrate the **`Regular_polygon`** class
    - Attach more than 4x4 regular polygons of different properties
    - Wait for next button
    - Detach all of the regular polygons
  - Demonstrate the **`Star`** class
    - Attach more than 4x4 stars of different properties
    - Wait for next button
    - Detach all of the stars

Basic Requirements

- Screenshots





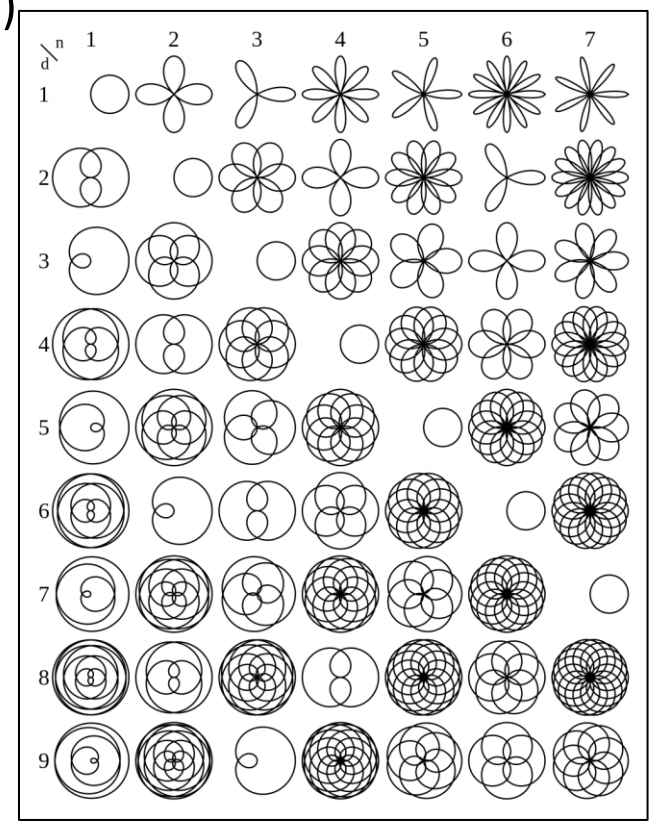


Advanced Requirements

- **Rose**
  - Center position (stored in **`Shape::points`**)
  - Numerator and denominator (**n, d** > 0, **`int`**s)
  - Radius (**r**, **`double`**)

- **How to draw**
  - $x = r\cos(k\theta)\cos(\theta)$
  - $y = r\cos(k\theta)\sin(\theta)$

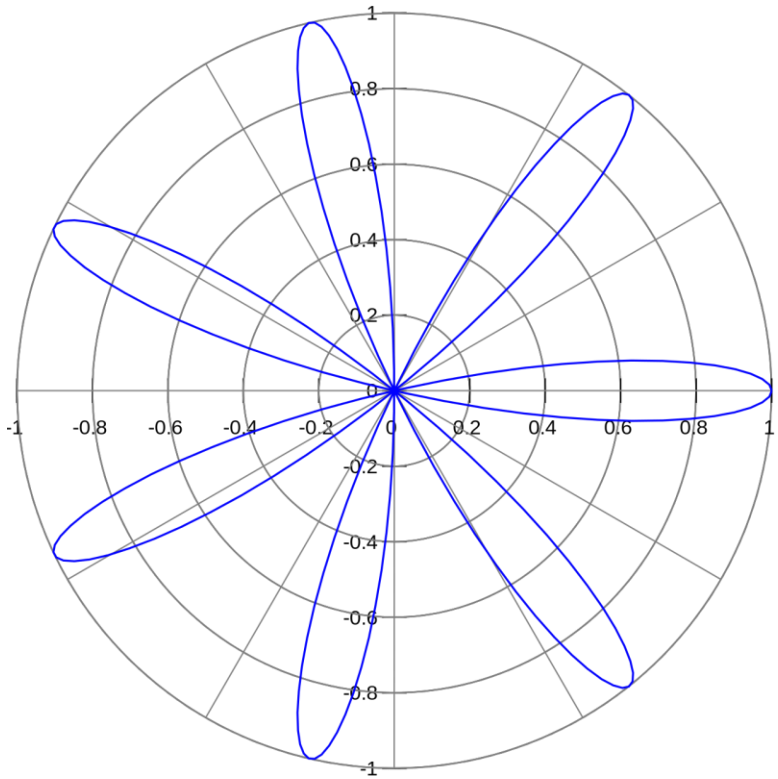
  - $k = n/d$
  - $0 \le \theta \le \pi dp$
  - $p = \begin{cases} 1 & \text{if } nd \text{ is odd} \\ 2 & \text{if } nd \text{ is even} \end{cases}$



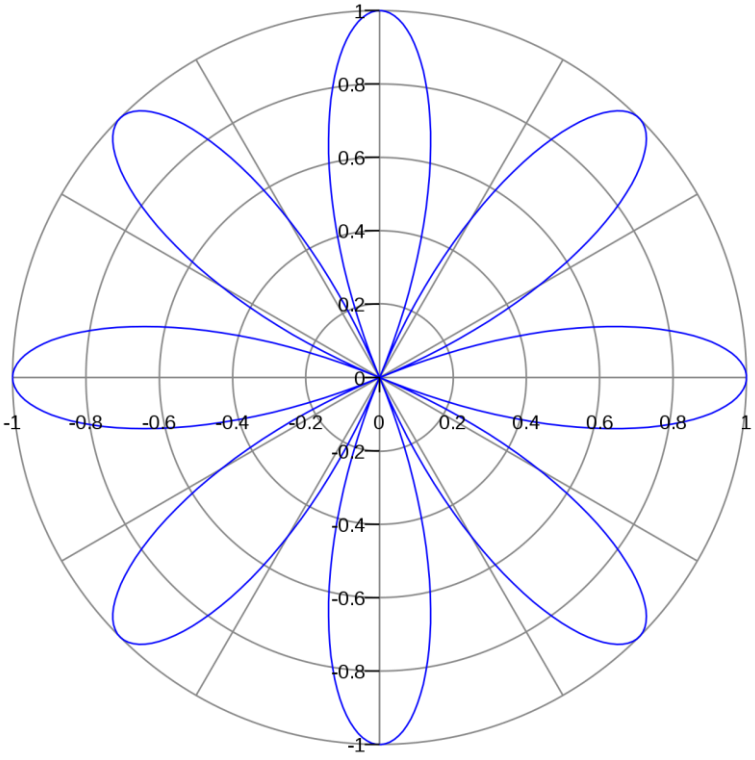
https://en.wikipedia.org/wiki/Rose_(mathematics)

Advanced Requirements

- **Rose**



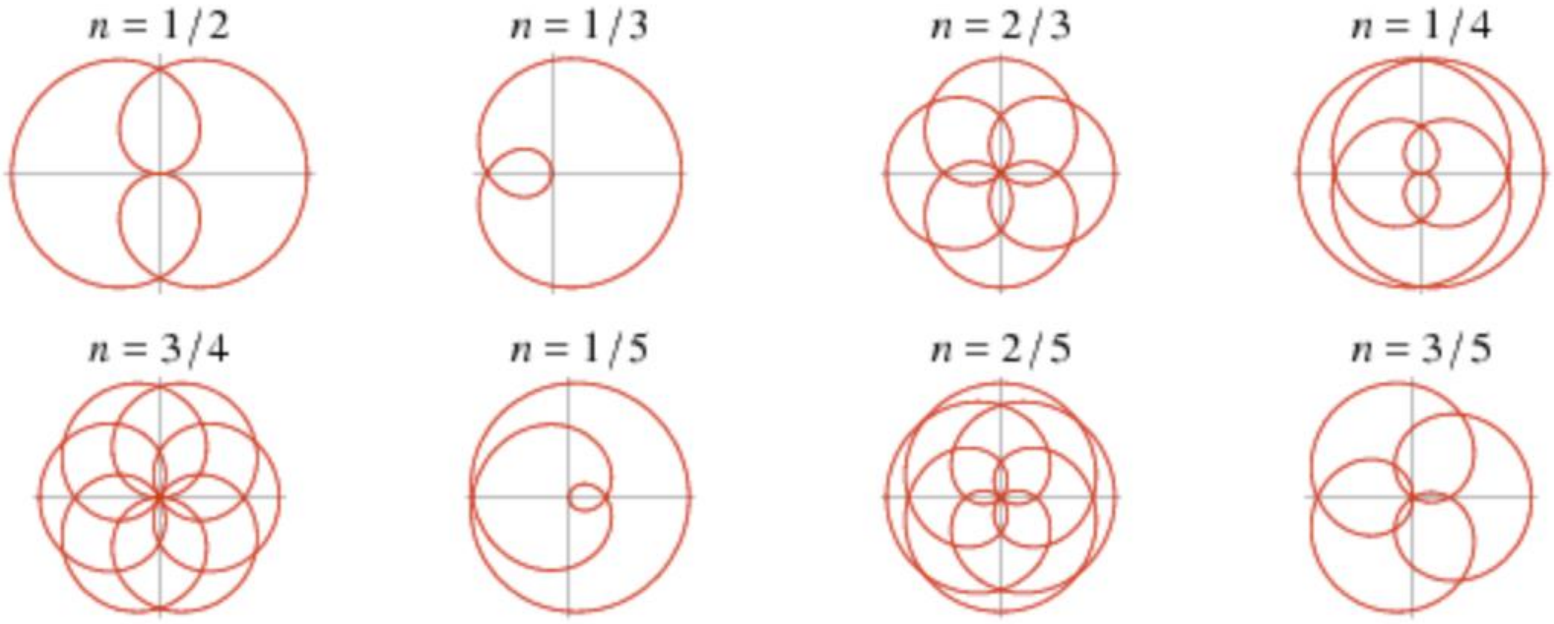
$k = 7$



$k = 4$

Advanced Requirements

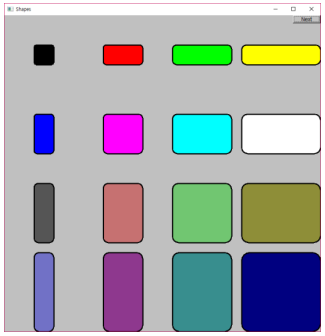
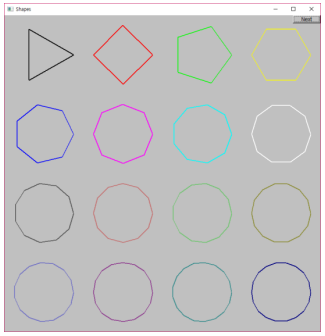
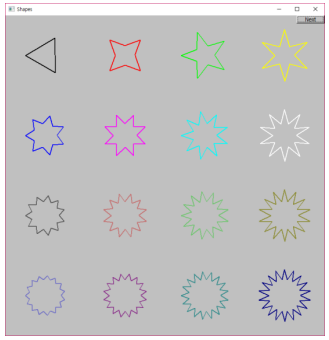
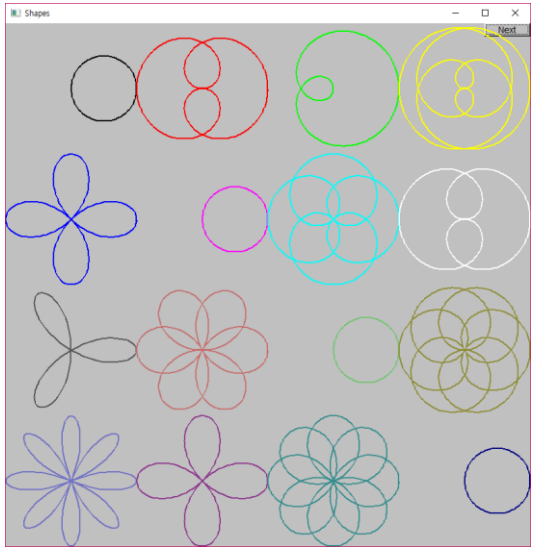
- **Rose**



https://mathworld.wolfram.com/Rose.html

Additional Requirements

- **Extend main_shapes.cpp**
  - Demonstrate the **Rose** class
    - Attach more than 4x4 roses of different properties
    - Wait for next button
    - Detach all of the roses

- **(Optional) add an additional mathematically defined shapes freely**
  - https://en.wikipedia.org/wiki/List_of_mathematical_shapes

Note

- **Report (*.pdf)**
  - Title page
    - Course title, submission date, affiliation, student ID, full name
  - Begin with a summary of your results
    - Which requirements did you fulfill? And which didn't you? (present a simple table)
    - Did you implement some additional features? What are those?
  - For each requirement (basic/advanced/optional), explain how you fulfilled it
    - Do not just dump the entire code
    - It's okay to copy snippets of your code to complement written description
  - Conclude with some comments on your work
    - Key challenges you have successfully tackled
    - Limitations you hope to address in the future

Submission

- **Compress your code and report into a single *.zip file**
  - Code
    - The entire project folder including *.sln, *.cpp, *.h, etc.
    - The grader should be able to open the *.sln with Visual Studio 2019 and build/run the project immediately without any problems
    - Remove Debug, Release, and .vs subfolders for compactness
  - Report
    - A single *.pdf file
    - You should convert your word format (*.hwp, *.doc, *.docx) to PDF format (*.pdf) before zipping
  - Name your zip file as your student ID
    - ex) 2012726055.zip
- **Upload to homework assignment in U-campus**
- **Due at 6/7 (Sun), 11:59 PM**