



Universidad del Istmo de Guatemala
Facultad de Ingenieria
Ing. en Sistemas
Informatica II
Prof. Ernesto Rodriguez - erodriguez@unis.edu.gt

Laboratorio #7

Fecha de entrega: 21 de Marzo, 2019 - 11:59pm

Instrucciones: Resolver cada uno de los ejercicios siguiendo sus respectivas instrucciones. El trabajo debe ser entregado a traves de Github, en su repositorio del curso, colocado en una carpeta llamada "Laboratorio #7". Al menos que la pregunta indique diferente, todas las respuestas a preguntas escritas deben presentarse en un documento formato pdf, el cual haya sido generado mediante Latex. Este laboratorio debe ser elaborado en parejas.

Tarea #1 (20%)

Defina una **clase virtual** llamada "Figura". Esta clase debe tener 3 **metodos virtuales**:

- "`double area() const`" el cual debe calcular el area de la figura
- "`void escalar(const double escala)`" el cual debe cambiar el tamaño de la figura proporcionalmente al parametro dado
- "`void mover(const Vector2d lugar)`" el cual debe desplazar la figura en la dirección indicada por el vector que se dio como parametro.
- "`bool estaAdentro(const Vector2d posicion) const`", el cual acepta un vector y retorna `true` si el vector esta adentro o `false` de lo contrario
- "`string toString() const`", el cual debe retornar una representación de la figura.

Tarea #2 (20%)

Definir la clase "Circulo" la cual debe implementar la **clase virtual** "Figura". Para representar el circulo debe utilizar un "Vector2d" el cual representa la posición del centro del circulo y un `double` para representar el radio del circulo. Debe tener un constructor que acepta el centro y radio como parametro. Utilizar el analisis matematico necesario para implementar los metodos de "Figura". Recuerde que el area de un circulo esta dado por $A_o = \pi * r^2$. El metodo "toString" debe representar el circulo en el siguiente formato: "`c(x,y,r)`".

Tarea #3 (20%)

Definir la clase “Rectangulo” la cual debe implementar la **clase virtual** “Figura”. Para representar el rectangulo se debe utilizar un “Vector2d” para indicar la posición de la esquina inferior del rectangulo, un double para indicar el ancho y otro double para indicar el largo del rectangulo. El metodo “toString” debe representar el rectangulo en el siguiente formato: “r(x,y,largo,ancho)”.

Tarea #4 (20%)

Definir la función “`const abarcar(Figura& figura, Vector2d[] puntos, int cantidad)`” la cual acepta una Figura como parametro y un arreglo con “cantidad” vectores de dos dimensiones. Este metodo debe mover y/o escalar la figura de tal manera que todos los puntos esten adentro de la Figura. Este metodo debe intentar que la figura tenga al menor area posible despues de la ejecución, siempre y cuando todos los puntos se encuentren dentro de la nueva figura.

Tarea #5 (20%)

Declarar un metodo “main” el cual debe hacer lo siguiente:

1. Solicitar al usuario un caracter, estos pueden ser “v” para vector, “c” para circulo o “r” para rectangulo.
2. Si el caracter es “v”:
 - (a) Solicitar dos numeros reales que corresponden a las coordenadas del vector
 - (b) Regresar al paso #1
3. Si el caracter es “c”:
 - (a) Solicitar dos numeros reales correspondiendo al vector del centro del circulo
 - (b) Solicitar un numero real correspondiente al radio del circulo
 - (c) Ir al paso #5
4. Si el caracter es “r”:
 - (a) Solicitar dos numeros reales correspondiendo al vector de la esquina inferior del rectangulo
 - (b) Solicitar un numero real correspondiendo al ancho del rectangulo
 - (c) Solicitar un numero real correspondiendo al largo del rectangulo
 - (d) Ir al paso #5
5. Llamar a la función “abarcar” con los vectores y figura que fueron ingresados en los pasos 1-4.
6. Imprimir la figura resultante y terminar