Nombre: Jeimy Marley Morales Sosa & Camilo Andrés Orrico Emme

ID: L00418832 //L

Materia: Estructura

Ejercicio 1 Codificación

Calcular área de las figuras

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* UFA-ESPE

\* Autores: Jeimy Marley Morales Sosa & Camilo Andres Orrico Emme

\* Modificacion: Martes, 08 de nombre de 2022 21:17:59

\* Purpose: Areas de las figuras

\*Nivel: Tercero NRC:7999

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include "area.cpp"

#include <iostream>

#include <stdlib.h>

#include <stdio.h>

#include <ctype.h>

#include <string.h>

#include <conio.h>

#include <windows.h>

using namespace std;

int main(int argc, char\*\* argv ){

HANDLE hConsole = GetStdHandle( STD\_OUTPUT\_HANDLE );

SetConsoleTextAttribute(hConsole, 3);

float altura,radio1,radio2;

double lado,base,c,r,t,o,ci;

Area \*obj= new Area(lado,base,altura,radio1,radio2,c,r,ci,o,t);

obj->ingresar(\*obj);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* UFA-ESPE

\* Autores: Jeimy Marley Morales Sosa & Camilo Andres Orrico Emme

\* Modificacion: Martes, 08 de nombre de 2022 21:17:59

\* Purpose: Areas de las figuras

\*Nivel: Tercero NRC:7999

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include "Area.h"

#include "Area.h"

#include "validar.cpp"

#include <clocale>

#include <conio.h>

#include <windows.h>

using namespace std;

float Area::getLado(void)

{

return lado;

}

void Area::setLado(float newLado)

{

lado = newLado;

}

double Area::getBase(void)

{

return base;

}

void Area::setBase(double newBase)

{

base = newBase;

}

float Area::getAltura(void)

{

return altura;

}

void Area::setAltura(float newAltura)

{

altura = newAltura;

}

float Area::getPI(void)

{

return PI;

}

void Area::setPI(float newPI)

{

PI = newPI;

}

int Area::getRadio1(void)

{

return radio1;

}

void Area::setRadio1(int newRadio1)

{

radio1 = newRadio1;

}

float Area::getRadio2(void)

{

return radio2;

}

void Area::setRadio2(float newRadio2)

{

radio2 = newRadio2;

}

double Area::getCuadrado(void)

{

return cuadrado;

}

void Area::setCuadrado(double newCuadrado)

{

cuadrado = newCuadrado;

}

double Area::getRectangulo(void)

{

return rectangulo;

}

void Area::setRectangulo(double newRectangulo)

{

rectangulo = newRectangulo;

}

double Area::getCirculo(void)

{

return circulo;

}

void Area::setCirculo(double newCirculo)

{

circulo = newCirculo;

}

double Area::getOvalo(void)

{

return ovalo;

}

void Area::setOvalo(double newOvalo)

{

ovalo = newOvalo;

}

double Area::getTriangulo(void)

{

return triangulo;

}

void Area::setTriangulo(double newTriangulo)

{

triangulo = newTriangulo;

}

Area::Area(float lado, double base, float altura, int radio1, float radio2, double cuadrado, double rectangulo, double circulo, double ovalo, double triangulo)

{

this->base=base;

this->lado=lado;

this->altura=altura;

this->radio1=radio1;

this->radio2=radio2;

this->cuadrado=cuadrado;

this->rectangulo=rectangulo;

this->circulo=circulo;

this->ovalo=ovalo;

this->triangulo=triangulo;

}

Area::~Area()

{

// TODO : implement

}

void Area::ingresar(Area obj)

{

setlocale(LC\_CTYPE,"Spanish");

int figura,base1,radio;

float area, base,altura,radio1;

char datoEntero[10],datoReal[10];

do{

cout<<"\nCALCULA EL AREA DE LAS SIGUIENTES FIGURAS"<<endl;

cout<<"\n[1] Cuadrado"<<"\n[2] Triangulo"<<"\n[3] Rectangulo"<<"\n[4] circulo"<<"\n[5] Ovalo"<<"\n[0] Salir"<<endl;

str\_cpy(datoEntero,ingresarDatosEnteros("\n Ingrese el valor: "));

figura=funcion\_atoi(datoEntero,strlen(datoEntero));

if(figura<=5 && figura >=1){

cout<<"\nINGRESO DATO VALIDO"<<endl;

}

if(figura==0) {

break;

}

system("cls");

switch(figura){

case 1:

cout<<"\n~~~~~~~~~~Calculár el area de un cuadrado~~~~~~~~~~~~~~~~~~"<<endl;

str\_cpy(datoReal,ingresoflotantes("REAL= Ingrese el valor del lado del cudrado: "));

area=funcion\_strtod(datoReal);

obj.setLado(area);

obj.calcularArea(area);

cout<<"\nArea del cuadrado es: "<<obj.getCuadrado()<<" cm ^2"<<endl;

break;

case 2:

cout<<"\n~~~~~~~~~~Calculár el area de un triangulo~~~~~~~~~~~~~~~~~~"<<endl;

str\_cpy(datoReal,ingresoflotantes("\nREAL= Ingrese la altura del triangulo: "));

altura=funcion\_strtod(datoReal);

obj.setAltura(altura);

str\_cpy(datoReal,ingresoflotantes("\nREAL= Ingrese la base del triangulo: "));

base=funcion\_strtod(datoReal);

obj.setBase(base);

obj.calcularArea(base,altura);

cout<<"\nArea del triangulo es:"<<obj.getTriangulo()<<" cm"<<endl;

break;

case 3:

cout<<"\n~~~~~~~~~~Calculár el area de un rectangulo~~~~~~~~~~~~~~~~~~"<<endl;

cout<<"\nAltura valor real base valor entero"<<endl;

str\_cpy(datoReal,ingresoflotantes("REAL= Ingrese la altura del rectangulo: "));

altura=funcion\_strtod(datoReal);

obj.setAltura(altura);

str\_cpy(datoEntero,ingresarDatosEnteros("\nENTERO= Ingrese la base del rectangulo: "));

base1=funcion\_atoi(datoEntero,strlen(datoEntero));

obj.setBase(base1);

obj.calcularArea(altura,base1);

cout<<"\nArea del rectangulo es: "<<obj.getRectangulo()<<endl;

break;

case 4:

cout<<"\n~~~~~~~~~~Calculár el area de un circulo~~~~~~~~~~~~~~~~~~"<<endl;

str\_cpy(datoEntero,ingresarDatosEnteros("\nENTERO= Ingrese el radio del circulo: "));

radio=funcion\_atoi(datoEntero,strlen(datoEntero));

obj.setRadio1(radio);

obj.calcularArea(radio);

cout<<"\nArea del circulo es: "<<obj.getCirculo()<<endl;

break;

case 5:

cout<<"\n~~~~~~~~~~Calculár el area de un ovalo~~~~~~~~~~~~~~~~~~"<<endl;

str\_cpy(datoEntero,ingresarDatosEnteros("\nENTERO= Ingrese el primer radio del ovalo: "));

radio=funcion\_atoi(datoEntero,strlen(datoEntero));

obj.setRadio1(radio);

str\_cpy(datoReal,ingresoflotantes("\nREAL= Ingrese el segundo radio del ovalo: "));

radio1=funcion\_strtod(datoReal);

obj.setRadio2(radio1);

obj.calcularArea(radio,radio1,3.1316);

cout<<"\nArea del ovalo es: "<<obj.getOvalo()<<endl;

break;

}

return ingresar(obj);

}while(figura>5 );

}

double Area::calcularArea(int radio)

{

double area;

area=(radio\*radio)\*3.1416;

this->setCirculo(area);

return area;

}

double Area::calcularArea (double lado){//cuadrado

double area;

area=lado\*lado;

this->setCuadrado(lado\*lado);

return lado\*lado;

}

double Area::calcularArea(float base, float altura){//triangulo

float area;

area=(base\*altura)/2;

this->setTriangulo(area);

return area;

}

double Area::calcularArea(float altura, int base){//rectangulo

float area;

area=altura\*base;

this->setRectangulo(area);

return area;

}

double Area::calcularArea(int radio1, float radio2,float pi){//ovalo

float area;

area=radio1\*radio2\*pi;

this->setOvalo(area);

return area;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* UFA-ESPE

\* Autores: Jeimy Marley Morales Sosa & Camilo Andres Orrico Emme

\* Modificacion: Martes, 08 de nombre de 2022 21:17:59

\* Purpose: Areas de las figuras

\*Nivel: Tercero NRC:7999

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

#include <stdio.h>

#include <string.h>

#include <conio.h>

using namespace std;

int exponente(int base, int exponent);

int funcion\_atoi(char \*str,size\_t len);

double funcion\_strtod(char \*s);

char \*ingresarDatosEnteros(char const \*msj);

char \*str\_cpy(char \*cadena1 ,char \*cadena2);//copia la cadena dos en la cadena uno

char \*ingresocaracteres(char \*msj);

char \*ingresoflotantes(char const \*msj);

//Funcion pow

int exponente(int base, int exponent){

if (exponent == 0) {

return 1;

} else {

int result = base;

for (int i = 1; i < exponent; i++) {

result \*= base;

}

return result;

}

}

//Funcion atoi combierte a entero

int funcion\_atoi(char \*str,size\_t len){

int i;

int res;

int signo;

i = 0;

res = 0;

signo = 1;

while(str[i] == ' ' || (str[i] >= 9 && str[i] <= 13))

{

i++;

}

if(str[i] == '-')

{

signo = -1;

i++;

}

else if(str[i] == '+')

{

signo = 1;

i++;

}

int j = i-1;

i = len;

while (str[i] < '0' || str[i] > '9') {

i--;

}

int num\_final = i;

while(i > j)

{

if (str[i] >= '0' && str[i] <= '9') {

res += exponente(10, num\_final-i) \* (str[i] - '0');

} else {

num\_final--;

}

i--;

}

return(res \* signo);

}

char \*str\_cpy(char \*cadena1 ,char \*cadena2){

if(cadena1 == NULL) return NULL;

int i=0;

while (cadena2[i] !='\0')

{

cadena1[i]=cadena2[i];

i++;

}

cadena1[i]='\0';

return cadena1;

}

double funcion\_strtod(char \*s)

{

int count=0;

int i;

for(i=0; i<=strlen(s)-1; i++)

{

if(s[i]=='.')

{

count++;

}

if(count==2)

{

return -1;

}

if( !((s[i]>=48 && s[i]<=57)|| (s[i]=='.')) )

{

return -1;

}

}

if(count==0)

{

return -1;

}

double d;

sscanf(s,"%lf",&d);

return d;

}

/////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

char \*ingresoflotantes(char const \*msj)

{

char \*dato=new char[50];

char c;

int sum=0;

int i=0;

printf("%s",msj);

while (((c=getch())!=13)&& sum<=1){

if ((c>='0'&&c<='9') || c=='.'){

printf("%c",c);

dato[i++]=c;

if (c=='.'){

sum++;

if (sum>=2){

putchar(8);

putchar(32);

putchar(8);

sum--;

}

}

}else if(c==8){

dato[i]=0;

i--;

putchar(8);

putchar(32);

putchar(8);

}

}

dato[i]='\0';//fin del while

return dato;

}

char \*ingresocaracteres(char \*msj)

{

char \*dato=new char[10];

char c;

int i=0;

printf("%s",msj);

while ((c=getch())!=13){

if((c>='a'&& c<='z')||(c>='A'&&c<='Z')){

printf("%c",c);

dato[i++]=c;

}else if(c==8){

dato[i]=0;

i--;

putchar(8);

putchar(32);

putchar(8);

}

}

dato[i]='\0';//fin del while

return dato;

}

////clase

char \*ingresarDatosEnteros(char const \*msj){

char \*dato=new char[10];

char c;

int i=0;

printf("%s",msj);

while((c=getch())!=13){

if(c>='0'&& c<='9'){

printf("%c",c);

dato[i++]=c;

}else if(c==8){

dato[i]=0;

i--;

putchar(8);

putchar(32);

putchar(8);

}

}

dato[i]='\0';

return dato;

}

template<typename T>

class Datos{

public:

void setData(T d);

T getData() const;

private:

T data;

};

template<typename T>

void Datos<T>::setData(T d){

this->data=d;

}

template<typename T>

T Datos<T>::getData()const{

return this->data;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* UFA-ESPE

\* Autores: Jeimy Marley Morales Sosa & Camilo Andres Orrico Emme

\* Modificacion: Martes, 08 de nombre de 2022 21:17:59

\* Purpose: Areas de las figuras

\*Nivel: Tercero NRC:7999

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#if !defined(\_\_Class\_Diagram\_1\_Area\_h)

#define \_\_Class\_Diagram\_1\_Area\_h

class Area

{

public:

float getLado(void);

void setLado(float newLado);

double getBase(void);

void setBase(double newBase);

float getAltura(void);

void setAltura(float newAltura);

float getPI(void);

void setPI(float newPI);

int getRadio1(void);

void setRadio1(int newRadio1);

float getRadio2(void);

void setRadio2(float newRadio2);

double getCuadrado(void);

void setCuadrado(double newCuadrado);

double getRectangulo(void);

void setRectangulo(double newRectangulo);

double getCirculo(void);

void setCirculo(double newCirculo);

double getOvalo(void);

void setOvalo(double newOvalo);

double getTriangulo(void);

void setTriangulo(double newTriangulo);

Area(float lado, double base, float altura, int radio1, float radio2, double cuadrado, double rectangulo, double circulo, double ovalo, double triangulo);

~Area();

void ingresar(Area obj);

double calcularArea(int radio);//circulo

double calcularArea(float base, float altura);//triangulo

double calcularArea(float altura, int base);//rectangulo

double calcularArea(int radio1, float radio2,float pi);//ovalo

double calcularArea(double lado);//cuadrado

protected:

private:

float lado;

double base;

float altura;

float PI;

int radio1;

float radio2;

double cuadrado;

double rectangulo;

double circulo;

double ovalo;

double triangulo;

};

#endif

**Ejercicio 2 Codificación**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* UFA-ESPE

\* Autores: Jeimy Marley Morales Sosa & Camilo Andres Orrico Emme

\* Modificacion: Martes, 08 de nombre de 2022 21:17:59

\* Purpose: Areas de las figuras

\*Nivel: Tercero NRC:7999

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include "SobrecargaOperador.cpp"

#include <iostream>

#include <stdlib.h>

#include <stdio.h>

#include <ctype.h>

#include <string.h>

#include <conio.h>

#include <windows.h>

using namespace std;

int main(int argc, char\*\* argv ){

HANDLE hConsole = GetStdHandle( STD\_OUTPUT\_HANDLE );

SetConsoleTextAttribute(hConsole, 3);

float cuadrado, lado, triangulo, altura,base, rectangulo, circulo, radio;

SobrecargaOperador \*obj= new SobrecargaOperador(cuadrado,lado,triangulo,altura, base,rectangulo,circulo,radio);

obj->ingresar(\*obj);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* UFA-ESPE

\* Autores: Jeimy Marley Morales Sosa & Camilo Andres Orrico Emme

\* Modificacion: Martes, 08 de nombre de 2022 21:17:59

\* Purpose: Areas de las figuras

\*Nivel: Tercero NRC:7999

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include "SobrecargaOperador.h"

#include <windows.h>

#include "validar.cpp"

using namespace std;

float SobrecargaOperador::getCuadrado(void)

{

return cuadrado;

}

void SobrecargaOperador::setCuadrado(float newCuadrado)

{

cuadrado = newCuadrado;

}

float SobrecargaOperador::getLado(void)

{

return lado;

}

void SobrecargaOperador::setLado(float newLado)

{

lado = newLado;

}

float SobrecargaOperador::getTriangulo(void)

{

return triangulo;

}

void SobrecargaOperador::setTriangulo(float newTriangulo)

{

triangulo = newTriangulo;

}

float SobrecargaOperador::getBase(void)

{

return base;

}

void SobrecargaOperador::setBase(float newBase)

{

base = newBase;

}

float SobrecargaOperador::getAltura(void)

{

return altura;

}

void SobrecargaOperador::setAltura(float newAltura)

{

altura = newAltura;

}

float SobrecargaOperador::getRectangulo(void)

{

return rectangulo;

}

void SobrecargaOperador::setRectangulo(float newRectangulo)

{

rectangulo = newRectangulo;

}

float SobrecargaOperador::getCirculo(void)

{

return circulo;

}

void SobrecargaOperador::setCirculo(float newCirculo)

{

circulo = newCirculo;

}

float SobrecargaOperador::getRadio(void)

{

return radio;

}

void SobrecargaOperador::setRadio(float newRadio)

{

radio = newRadio;

}

SobrecargaOperador::SobrecargaOperador(float cuadrado, float lado, float triangulo,float altura, float base, float rectangulo, float circulo, float radio)

{

// TODO : implement

}

SobrecargaOperador::~SobrecargaOperador()

{

// TODO : implement

}

SobrecargaOperador SobrecargaOperador::calcular\_cuadrado(SobrecargaOperador obj)

{

SobrecargaOperador Obj1=obj\*obj;

this->setCuadrado(obj.getCuadrado());

return \*this;

}

SobrecargaOperador SobrecargaOperador::calculartriangulo(SobrecargaOperador obj)

{

SobrecargaOperador Obj1=obj\*obj;

this->setTriangulo(obj.getTriangulo());

return \*this;

}

SobrecargaOperador SobrecargaOperador::calcularCirculo(SobrecargaOperador obj)

{

SobrecargaOperador Obj1=obj\*obj;

this->setCirculo(obj.getCirculo());

return \*this;

}

SobrecargaOperador SobrecargaOperador::calcularRectangulo(SobrecargaOperador obj)

{

SobrecargaOperador Obj1=obj\*obj;

this->setRectangulo(obj.getRectangulo());

return \*this;

}

void SobrecargaOperador::ingresar(SobrecargaOperador obj){

int figura,base1,radio;

float lado, base,altura,radio1;

char datoEntero[10],datoReal[10];

do{

cout<<"\nCALCULA EL AREA DE LAS SIGUIENTES FIGURAS"<<endl;

cout<<"\n[1] Cuadrado"<<"\n[2] Triangulo"<<"\n[3] Rectangulo"<<"\n[4] circulo"<<"\n[0] Salir"<<endl;

str\_cpy(datoEntero,ingresarDatosEnteros("\n Ingrese el valor: "));

figura=funcion\_atoi(datoEntero,strlen(datoEntero));

if(figura<=5 && figura >=1){

cout<<"\nINGRESO DATO VALIDO"<<endl;

}

if(figura==0) {

break;

}

system("cls");

switch(figura){

case 1:

cout<<"\n~~~~~~~~~~Calculár el area de un cuadrado~~~~~~~~~~~~~~~~~~"<<endl;

str\_cpy(datoReal,ingresoflotantes("REAL= Ingrese el valor del lado del cudrado: "));

lado=funcion\_strtod(datoReal);

obj.setLado(lado);

obj.calcular\_cuadrado(obj);

cout<<"\nArea del cuadrado es: "<<obj.getCuadrado()<<" cm ^2"<<endl;

break;

case 2:

cout<<"\n~~~~~~~~~~Calculár el area de un triangulo~~~~~~~~~~~~~~~~~~"<<endl;

str\_cpy(datoReal,ingresoflotantes("\nREAL= Ingrese el valor de la altura del triangulo: "));

altura=funcion\_strtod(datoReal);

obj.setAltura(altura);

str\_cpy(datoReal,ingresoflotantes("\nREAL= Ingrese el valor de la base del triangulo: "));

base=funcion\_strtod(datoReal);

obj.setBase(base);

obj.calculartriangulo(obj);

cout<<"\nArea del triangulo es: "<<obj.getTriangulo()<<" cm "<<endl;

break;

case 3:

cout<<"\n~~~~~~~~~~Calculár el area de un rectangulo~~~~~~~~~~~~~~~~~~"<<endl;

str\_cpy(datoReal,ingresoflotantes("\nREAL= Ingrese el valor de la altura del rectangulo: "));

altura=funcion\_strtod(datoReal);

obj.setAltura(altura);

str\_cpy(datoReal,ingresoflotantes("\nREAL= Ingrese el valor de la base del rectangulo: "));

base=funcion\_strtod(datoReal);

obj.setBase(base);

obj.calcularRectangulo(obj);

cout<<"\nArea del Rectangulo es: "<<obj.getRectangulo()<<" cm "<<endl;

break;

case 4:

cout<<"\n~~~~~~~~~~Calculár el area de un circulo~~~~~~~~~~~~~~~~~~"<<endl;

str\_cpy(datoReal,ingresoflotantes("REAL= Ingrese el radio del circulo "));

radio=funcion\_strtod(datoReal);

obj.setRadio(radio);

obj.calcularCirculo(obj);

cout<<"\nArea del circulo es: "<<obj.getCirculo()<<endl;

break;

}

return ingresar(obj);

}while(figura>4 );

}

void SobrecargaOperador::imprimir(SobrecargaOperador obj){

}

SobrecargaOperador SobrecargaOperador::operator \*(SobrecargaOperador obj){

this->circulo=obj.getRadio()\*obj.getRadio()\*3.14;

this->cuadrado=obj.getLado()\*obj.getLado();

this->triangulo=( obj.getBase()\*obj.getAltura())\*0.5;

this->rectangulo=obj.getBase()\*obj.getAltura();

return \*this;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* UFA-ESPE

\* Autores: Jeimy Marley Morales Sosa & Camilo Andres Orrico Emme

\* Modificacion: Martes, 08 de nombre de 2022 21:17:59

\* Purpose: Areas de las figuras

\*Nivel: Tercero NRC:7999

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#if !defined(\_\_Class\_Diagram\_1\_SobrecargaOperador\_h)

#define \_\_Class\_Diagram\_1\_SobrecargaOperador\_h

class SobrecargaOperador

{

public:

float getCuadrado(void);

void setCuadrado(float newCuadrado);

float getLado(void);

void setLado(float newLado);

float getTriangulo(void);

void setTriangulo(float newTriangulo);

float getBase(void);

void setBase(float newBase);

float getAltura(void);

void setAltura(float newAltura);

float getRectangulo(void);

void setRectangulo(float newRectangulo);

float getCirculo(void);

void setCirculo(float newCirculo);

float getRadio(void);

void setRadio(float newRadio);

SobrecargaOperador(float cuadrado, float lado, float triangulo,float altura, float base, float rectangulo, float circulo, float radio);

~SobrecargaOperador();

SobrecargaOperador calcular\_cuadrado(SobrecargaOperador obj);

SobrecargaOperador calculartriangulo(SobrecargaOperador obj);

SobrecargaOperador calcularCirculo(SobrecargaOperador obj);

SobrecargaOperador calcularRectangulo(SobrecargaOperador obj);

void ingresar(SobrecargaOperador obj);

void imprimir(SobrecargaOperador obj);

SobrecargaOperador operator \*(SobrecargaOperador obj);

protected:

private:

float cuadrado;

float lado;

float triangulo;

float base;

float altura;

float rectangulo;

float circulo;

float radio;

};

#endif