



EXAMEN DE SISTEMAS INFORMÁTICOS INDUSTRIALES (SOLUCIÓN PRÁCTICAS)

La duración del examen es de 2 horas.

JUNIO 2016

1. Escribir un programa en C++ y C, que mediante el uso de clases y métodos, pida al usuario seleccionar una figura geométrica (círculo, triángulo o cuadrado), a continuación solicite sus medidas y finalmente permita calcular su área y perímetro.
 - a. Se facilita el fichero *clase.h* (que no debe modificarse) con la definición de la clase y los métodos a implementar.
 - b. Se facilita un main en el fichero *examen.cpp* que deberéis completar para que cumpla su función:
 - i. Deberéis añadir las líneas necesarias para que el menú se ejecute de forma continua tras seleccionar cada opción hasta que se marque la opción 0.
 - ii. Deberéis añadir las llamadas a los métodos en cada apartado.
 - c. Se facilita la estructura del fichero *clase.cpp* para que lo completéis con el constructor por defecto y destructor, así como los métodos a implementar:
 - i. Constructor por defecto:
 1. Deberán inicializarse las variables de la clase.
 - ii. Destructor:
 1. Deberá liberarse la memoria en caso de ser necesario.
 - iii. Método introducir_datos:
 1. Este método recibe la figura geométrica seleccionada pero sólo podrá ejecutarse si no hay una previamente cargada.
 2. Deberá reservar dinámicamente la memoria para almacenar la información necesaria para cada figura geométrica: radio para el círculo, base y altura para el triángulo y para el cuadrado.
 3. Finalmente deberán pedirse dichos valores al usuario para almacenarlos en el vector.
 - iv. Método calcular_area
 1. Deberá devolver el área de la forma seleccionada.
 - v. Método calcular_perimetro
 1. Deberá devolver el perímetro de la forma seleccionada. En el caso del triángulo, al no tener toda la información sobre el tipo de triángulo deberá indicarse por pantalla que no puede calcularse y devolver -1 para que en el menú no se muestre dicho valor.
 - vi. Método liberar
 1. Deberá inicializar las variables para poder seleccionar una nueva forma y liberar la memoria reservada en caso de ser necesario.

(2.25 puntos)

Se valorará el funcionamiento del programa así como la claridad del código y los comentarios.



Escuela Politécnica Superior de Elche

Grado en Ingeniería Electrónica y Automática Industrial

Se facilita el código del fichero **clase.h** que no debe modificarse.

```
#ifndef __CLASE_H__
#define __CLASE_H__

class clase{
protected:
    int tipo;
    float *valores;
    int figura_seleccionada;
public:
    clase();//Constructor
    ~clase(void); //Destructor
    void introducir_datos(int t);
        // Recibe tipo de forma,
        // reserva dinamicamente la memoria
        // y solicita los datos
    float calcular_area(); // Metodo que calcula el área
        // de la forma seleccionada
    float calcular_perimetro(); // Metodo que calcula el
        // perimetro de la forma seleccionada
    void liberar();
};
#endif
__CLASE_CPP__
else
#endif
#endif
```

Se facilita el código del fichero **clase.cpp** que debe modificarse.

```
#define __CLASE_CPP__

#include <iostream>
#include <math.h>
#include "clase.h"
#include <cstdlib>
#include <stdio.h>
#include <string.h>
#include <locale.h>
#define pi (3.141592653589793)
using namespace std;

clase::clase() {
}

clase::~~clase(void)
{
}
```



Se facilita el código del fichero **examen.cpp** que el alumno debe completar.

```
#define __EXAMEN_CPP__
#include <iostream>
#include "clase.h"
#include <cstdlib>
#include <stdio.h>
#include <string.h>
#include <locale.h>
using namespace std;

int main(void){
    int opcion_menu, tipo_de_forma;
    float perimetro, area;
    clase miclase;

    printf("\n- MENU -\n-----\n");
    printf("1: Seleccionar forma\n");
    printf("2: Calcular area\n");
    printf("3: Calcular perimetro\n");
    printf("4: Liberar\n");
    printf("0: SALIR\n");
    printf("Introduzca opcion del menu: ");
    scanf("%d",&opcion_menu);
    switch(opcion_menu){
        case 1:
            printf("Introduzca tipo de forma a crear\n");
            printf(" (1:circulo, 2:triangulo, 3:rectangulo): ");
            scanf("%d",&tipo_de_forma);
            break;
        case 2:
            printf("Area de la forma seleccionada: %.2f\n",area);
            break;
        case 3:
            if (perimetro==-1)
                break;
            printf("Perimetro de la forma seleccionada:
%.2f\n",perimetro);
            break;
        case 4:
            printf("La memoria ha sido liberada y ya puede crear una
nueva forma.\n");
            break;
        default:
            break;
    }
}
```



Escuela Politécnica Superior de Elche

Grado en Ingeniería Electrónica y Automática Industrial

Ejemplo de funcionamiento:

<pre>examen.exe - MENU - ----- 1: Seleccionar forma 2: Calcular area 3: Calcular perimetro 4: Liberar 0: SALIR Introduzca opcion del menu: 1 Introduzca tipo de forma a crear (1:circulo, 2:triangulo, 3:rectangulo): 1 Introduzca el radio del circulo: 5 - MENU - ----- 1: Seleccionar forma 2: Calcular area 3: Calcular perimetro 4: Liberar 0: SALIR Introduzca opcion del menu: 2 Area de la forma seleccionada: 78.54 - MENU - ----- 1: Seleccionar forma 2: Calcular area 3: Calcular perimetro 4: Liberar 0: SALIR Introduzca opcion del menu: 3 Perimetro de la forma seleccionada: 31.42 - MENU - ----- 1: Seleccionar forma 2: Calcular area 3: Calcular perimetro 4: Liberar 0: SALIR Introduzca opcion del menu: 4 La memoria ha sido liberada y ya puede crear una nueva forma. - MENU - ----- 1: Seleccionar forma 2: Calcular area 3: Calcular perimetro 4: Liberar 0: SALIR Introduzca opcion del menu: 1 Introduzca tipo de forma a crear (1:circulo, 2:triangulo, 3:rectangulo): 2 Introduzca la base del triangulo: 4 Introduzca la altura del triangulo: 6 - MENU - ----- 1: Seleccionar forma 2: Calcular area 3: Calcular perimetro 4: Liberar 0: SALIR Introduzca opcion del menu: 2 Area de la forma seleccionada: 12.00</pre>	<pre>- MENU - ----- 1: Seleccionar forma 2: Calcular area 3: Calcular perimetro 4: Liberar 0: SALIR Introduzca opcion del menu: 3 No tenemos informacion sobre el tipo de triangulo que es. - MENU - ----- 1: Seleccionar forma 2: Calcular area 3: Calcular perimetro 4: Liberar 0: SALIR Introduzca opcion del menu: 4 La memoria ha sido liberada y ya puede crear una nueva forma. - MENU - ----- 1: Seleccionar forma 2: Calcular area 3: Calcular perimetro 4: Liberar 0: SALIR Introduzca opcion del menu: 1 Introduzca tipo de forma a crear (1:circulo, 2:triangulo, 3:rectangulo): 3 Introduzca la base del rectangulo: 5 Introduzca la altura del rectangulo: 9 - MENU - ----- 1: Seleccionar forma 2: Calcular area 3: Calcular perimetro 4: Liberar 0: SALIR Introduzca opcion del menu: 2 Area de la forma seleccionada: 45.00 - MENU - ----- 1: Seleccionar forma 2: Calcular area 3: Calcular perimetro 4: Liberar 0: SALIR Introduzca opcion del menu: 3 Perimetro de la forma seleccionada: 28.00 - MENU - ----- 1: Seleccionar forma 2: Calcular area 3: Calcular perimetro 4: Liberar 0: SALIR Introduzca opcion del menu: 0</pre>
--	---



Solución:

examen.cpp

```
#define __EXAMEN_CPP__
#include <iostream>
#include "clase.h"
#include <cstdlib>
#include <stdio.h>
#include <string.h>
#include <locale.h>
using namespace std;

int main(void) {
    int opcion_menu, tipo_de_forma;
    float perimetro, area;
    clase miclase;

do
{
    printf("\n- MENU -\n-----\n");
    printf("1: Seleccionar forma\n");
    printf("2: Calcular area\n");
    printf("3: Calcular perimetro\n");
    printf("4: Liberar\n");
    printf("0: SALIR\n");
    printf("Introduzca opcion del menu: ");
    scanf("%d", &opcion_menu);
    switch(opcion_menu) {
        case 1:
            printf("Introduzca tipo de forma a crear\n");
            printf(" (1:circulo, 2:triangulo, 3:rectangulo): ");
            scanf("%d", &tipo_de_forma);
            miclase.introducir_datos(tipo_de_forma);
            break;
        case 2:
            area=miclase.calcular_area();
            printf("Area de la forma seleccionada: %.2f\n", area);
            break;
        case 3:
            perimetro=miclase.calcular_perimetro();
            if (perimetro==-1)
                break;
            printf("Perimetro de la forma seleccionada: %.2f\n", perimetro);
            break;
        case 4:
            miclase.liberar();
            printf("La memoria ha sido liberada y ya puede crear una nueva forma.\n");
            break;
        default:
            break;
    }
} while(opcion_menu!=0);
}
```

clase.cpp

```
#define __CLASE_CPP__
#include <iostream>
#include <math.h>
#include "clase.h"
#include <cstdlib>
#include <stdio.h>
#include <string.h>
#include <locale.h>
#define pi (3.141592653589793)
using namespace std;

clase::clase() { // Constructor
    tipo=0;
    valores=NULL;
    figura_seleccionada=0;
}
```



Escuela Politécnica Superior de Elche

Grado en Ingeniería Electrónica y Automática Industrial

```
class::~class(void)
{
    if(valores!=NULL)
        delete []valores;
}

void class::introducir_datos(int t){ // Recibe tipo y reserva dinamicamente la memoria
if(figura_seleccionada==1){
    printf("Ya hay una forma seleccionada, liberela antes de seleccionar una
nueva.\n");
return;
}
tipo=t;
figura_seleccionada=1;
switch(tipo){
    case 1:
        valores=(float*)malloc(1*sizeof(float));
        printf("Introduzca el radio del circulo: ");
        scanf("%f",&valores[0]);
        break;
    case 2:
        valores=(float*)malloc(2*sizeof(float));
        printf("Introduzca la base del triangulo: ");
        scanf("%f",&valores[0]);
        printf("Introduzca la altura del triangulo: ");
        scanf("%f",&valores[1]);
        break;
    case 3:
        valores=(float*)malloc(2*sizeof(float));
        printf("Introduzca la base del rectangulo: ");
        scanf("%f",&valores[0]);
        printf("Introduzca la altura del rectangulo: ");
        scanf("%f",&valores[1]);
        break;
    default:
        printf("No ha introducido una forma correcta.\nVuelva a seleccionar la
opcion y marque 1, 2 o 3.\n");
        figura_seleccionada=0;
        break;
}
}
float class::calcular_area(){ // Metodo que calcula el area de la forma seleccionada
switch(tipo){
    case 1:
        return(pi*pow(valores[0],2));
        break;
    case 2:
        return((valores[0]*valores[1])/2);
        break;
    case 3:
        return(valores[0]*valores[1]);
        break;
}
}
float class::calcular_perimetro(){ // Metodo que calcula el perimetro de la forma
seleccionada
switch(tipo){
    case 1:
        return(2*pi*valores[0]);
        break;
    case 2:
        printf("No tenemos informacion sobre el tipo de triangulo que es.\n");
        return(-1);
        break;
    case 3:
        return(2*valores[0]+2*valores[1]);
        break;
}
}
void class::liberar(){
tipo=0;
figura_seleccionada=0;
if(valores!=NULL)
    delete []valores;
}
```