	Fundamentos de Programación		901
		MULTIPLES GRADOS		
71901020	Septiembre - 2023 Original	Duración: 120 min.	EXAMEN: Tipo A Mixto	Nacional - U.E. 1º Cuatrimestre
[] Material: Ninguno				Hoja 1 de 3

FUNDAMENTOS DE PROGRAMACION	MODELO ORIG.	CONV. EXTRAORD. 22/23
GRADO DE INFORMÁTICA y GRADO DE TECNOLOGÍAS DE LA INFORMACIÓN-ETSII-UNED		
MATERIAL AUTORIZADO: NINGUNO		DURACIÓN: 2 HORAS
INSTRUCCIONES:	1) El test debe ser contestado en la hoja de marcas. Sólo una de las cuatro respuestas posibles de cada pregunta es correcta. 2) El test es eliminatorio y aporta un 30% de la nota final. Son necesarias 8 preguntas correctas (6 con las prácticas aprobadas) para que se corrija el ejercicio. 3) Cada respuesta correcta 1 pto. Respuesta incorrecta, doble o en blanco: 0 pto. 4) Puede quedarse, si lo desea, con esta hoja de examen.	

CUESTIONES DE TEST

1. En el fragmento de programa de C+/-:

```
#include <Algo.h>
```

Algo es un identificador de un módulo ..

- A. de cualquier clase
- B. desarrollado para el programa
- C. de implementación
- D. de librería estándar

2. En C+/-, la sentencia *continue* finaliza la iteración en curso..

- A. sólo de bucles while y do
- B. sólo de bucles do
- C. de cualquier clase de bucle
- D. sólo de bucles while

3.¿Cuántas veces se muestra el mensaje "Pregunta Test" con el siguiente código C+/-?:

```
#include <stdio.h>

int main() {
    for(int x=-1; x<=10; x++){
        if(x < 5) {continue;}
        else {
            printf("Pregunta Test \n");    }}}

```

- A. 6 veces
- B. Ninguna vez
- C. 5 veces
- D. 7 veces

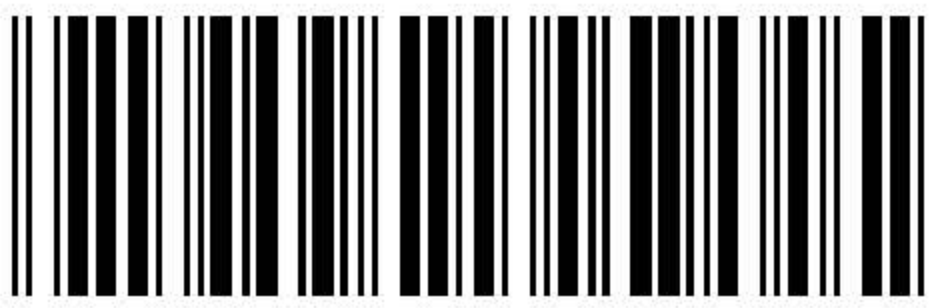

4.- En el fragmento de programa de C+/-:

```
while ( Suceso )
```

- A. Suceso es de tipo int
- B. Suceso es de tipo char
- C. Suceso es de tipo bool
- D. Suceso es de cualquier tipo predefinido

5.- La ordenación de un vector por inserción directa:

- A. sólo requiere un bucle
- B. requiere tres bucles
- C. requiere dos bucles
- D. no requiere ningún bucle

 71901020		Fundamentos de Programación		901
		MULTIPLES GRADOS		
Septiembre - 2023 Original	Duración: 120 min.	EXAMEN: Tipo A Mixto	Nacional - U.E. 1º Cuatrimestre	
Material: Ninguno			Hoja 2 de 3	

6.-La ejecución del siguiente código C+/- imprime:

```
#include <stdio.h>

int main(){
int i = 4;
do {
try {
if (i%2==0) {
throw 1;
} else {
throw 0;
}
} catch (int e) {
printf("%d ", e);
}
i=i-1;
} while (i > 0); }
```

- A. 0 1 0 1
- B. 1 1 1 0
- C. 1 0 1 1
- D. 1 0 1 0

7.- En el lenguaje C+/-, el fragmento de programa:

```
void Uno :: Dos (int Tres)
```

- A. Es la cabecera de un procedimiento
- B. Es la cabecera de una función
- C. Es un factor cualificado de una expresión
- D. Es una sentencia ejecutable de un subprograma

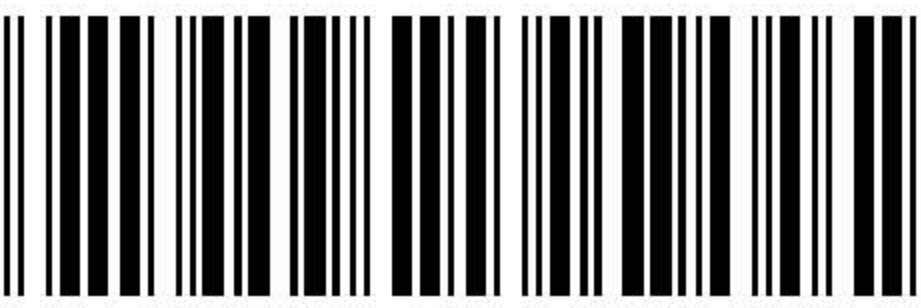

8.- En el lenguaje C+/-, la sentencia de programa:

```
Uno = new Dos;
```

- A. Uno es un puntero y Dos una variable
- B. Uno es un tipo de dato y Dos un puntero
- C. Uno es un puntero y Dos es un tipo de dato
- D. Uno es una variable y Dos un puntero

9.- ¿Cuál de las siguientes acciones no es un esquema general de la programación estructurada?

- A. Secuencia
- B. Ordenación
- C. Selección
- D. Iteración

 71901020		Fundamentos de Programación		901
		MULTIPLES GRADOS		
Septiembre - 2023 Original	Duración: 120 min.	EXAMEN: Tipo A Mixto	Nacional - U.E. 1º Cuatrimestre	
[] Material: Ninguno			Hoja 3 de 3	

10.- Una circunferencia se puede representar con el siguiente tipo C+/-:

```
typedef struct circunferencia {
    float centrox, centroy;
    float radio; };
```

¿Cuál sería el código del subprograma para saber si un punto (a,b) es externo a la circunferencia C ?

A.-

```
bool externo(circunferencia C, float a,float b) {
    return ((sqrt(powf(C.centrox-a,2)+powf(C.centroy-b,2)) > C.radio)); }
```

B.-

```
bool externo(circunferencia C, float a,float b) {
    return ((sqrt(powf(C.centrox-a,2)-powf(C.centroy-b,2)) > C.radio)); }
```

C.-

```
bool externo(circunferencia C, float a,float b) {
    return ((sqrt(powf(C.centrox-a,2)+powf(C.centroy-b,2)) < C.radio,2)); }
```

D.-

```
bool externo(circunferencia C, float a,float b) {
    return ((sqrt(powf(C.centrox-a,2)-powf(C.centroy-b,2)) < powf(C.radio,2))); }
```

EJERCICIO DE PROGRAMACIÓN

Realizar un TAD para gestionar información sobre un almacén de señales de tráfico. El almacén está organizado en secciones según el tipo de señales: Sección 1 para triangulares de peligro, Sección 2 para prioridad (forma variada como cedas el paso, stops,..), Sección 3 para circulares de prohibición y restricción, Sección 4 para circulares de obligación, y Sección 5 para rectangulares de indicación. Cada una de las cinco secciones dispone de espacio para un máximo de 100 tipos de señal y cada tipo de señal está identificado por un código de una letra y tres dígitos (por ejemplo, la señal de STOP es la R2 o el prohibido el paso es la R101), una descripción en texto de 24 caracteres, las dimensiones en alto, ancho y peso, y el número de señales disponibles en el almacén en un momento dado de ese tipo de señal.

El TAD debe incluir tres operaciones que son: **NuevoTipo**, **CargarEntrada** y **MostrarMenorNumero**. La operación **NuevoTipo** recibe la información necesaria de un nuevo tipo de señal y lo incluye en la sección que corresponda. La operación **CargarEntrada** recibe la información de la carga que un camión ha traído al almacén y actualiza el número de señales en el almacén: cada carga que entra tiene un máximo de ocho tipos de señales y son siempre tipos de señales ya existentes en el almacén. Y la operación **MostrarMenorNumero** muestra la sección, el código y la descripción de todos aquellos tipos de señales en las que hay menos de cinco unidades en el almacén.