

**Nombre:**

**Apellidos:**

**Grupo:**  A  B  C  D  E  F

**Advertencias:**

1. Duración del examen 1 horas y 30 minutos
2. No desgrape el cuadernillo del examen.
3. Puede utilizar lápiz o bolígrafo indistintamente.
4. No puede utilizar exit, continue, ni break (salvo en la instrucción switch()).

**Calificación:**

Código 1 (2.5 puntos)	Código 2 (2.5 puntos)	Código 3 (2 puntos)	Código 4 (3 puntos)	Total

### Código 1: Programa completo (2.5 puntos)

Escribir un programa completo (cabecera, includes, ...) que realice las siguientes operaciones:

- Declarar una variable x (tipo double) y otra n (tipo int).
- Leer el valor de ambas variables mediante scanf
- Calcular  $x^n$  a base de multiplicar x tantas veces como indique n.
- Mostrar el resultado utilizando un formato que sólo muestre 2 cifras en la parte decimal.

```
/*
Fecha: 20/nov/2008
Descripción: Para calcular la potencia de un número real.
*/
```

```
#include <stdio.h>
```

```
int main(void)
{
```

```
    double x; //base
    int n;    //exponente
    double res; //resultado
    int i;    //contador
```

```
    /*** Lectura ***/
```

```
    printf("Valor de la base: ");
    scanf("%lf", &x);
```

```
    do {
```

```
        printf("Valor del exponente: ");
        scanf("%d", &n);
```

```
        if (n<0) { //no conderamos exponentes negativos
            printf("Debe ser positivo. \n");
```

```
        }
```

```
    } while (n<0);
```

```
    /*** Calculo ***/
```

```
    res=1; //Esta inicialización es parte del cálculo
    for(i=0; i<n; i++) {
        res*=x;
    }
```

```
    /*** Mostrar el resultado ***/
```

```
    printf("%.2f ^ %d = %.2f", x, n, res);
```

```
    return 0;
```

```
}
```

```
/*
```

Comentarios:

Este programa no admite exponente negativo, aunque sería sencillo adaptarlo  
 Aunque el exponente sea 0 funciona bien porque el resultado es 1

```
*/
```

## Código 2: Programa completo (2.5 puntos)

Escribir un programa que pida una nota de un alumno (número real) y escriba las calificaciones correspondientes de acuerdo con el siguiente criterio:

0 a < 5.0          Suspenso  
5 a < 7.0          Aprobado  
7.0 a < 9.0        Notable  
9.0 a <= 10        Sobresaliente

Si la nota es negativa o mayor que 10 se debe mostrar un mensaje de error.

```
/*  
Nombre:  
Fecha: 20/nov/2008  
Descripción: Convertir una nota a formato texto  
*/  
  
#include <stdio.h>  
  
int main(void)  

```

### Código 3: Bucle (2 puntos)

Suponiendo que un programa ha inicializado, mediante #define el parámetro SECRETO con un valor entero, escribir un bucle que permita al usuario jugar a adivinar el valor de SECRETO. El bucle debe hacer:

- Preguntar un número al usuario
- Comparar dicho número con el valor de SECRETO
  - Si ha acertado, mostrar un mensaje de enhorabuena
  - Si no ha acertado, indicar si el valor SECRETO es menor o mayor que el introducido
- El bucle se repite hasta que el usuario acierte el valor

**Nota:** No hace falta validar la entrada, el usuario puede introducir cualquier valor entero y el programa se limita a darle pistas.

```

/*
Nombre:
Fecha: 20/nov/2008
Descripción: Programa para adivinar un número entero.
*/

#include <stdio.h>

#define SECRETO 1234

int main(void)
{
    int num; //número que prueba el usuario

    /** Bienvenida **/
    printf("Hola, a ver si adivinas el número secreto\n\n");

    /** Bucle para adivinar el número **/
    do {
        printf("Valor: ");
        scanf("%d", &num);
        if (num==SECRETO) {
            printf("Enhorabuena!!!\n");
            printf("Has acertado el número secreto\n");
        } else {
            if (num<SECRETO) printf("El número secreto es mayor\n");
            else printf("El número secreto es menor\n");
        }
    } while(num!=SECRETO);

    printf("Adios.\n");
    return 0;
}

```

### Código 4: Condiciones if (3 puntos)

Suponiendo que un programa ha leído una fecha mediante la variable entera `di` y la variable entera `mes`, indicar si es primavera, verano, otoño o invierno.

**Datos:** Las fechas de cambio de estación son:

20/mar Primavera

21/jun Verano

22/sep Otoño

21/dic Invierno

(Fechas de 2009, <http://en.wikipedia.org/wiki/Equinox>)

/\*

Nombre:

Fecha: 20/nov/2008

Descripción: Programa para identificar la estación del año.

\*/

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int di; //día del mes
```

```
    int mes;
```

```
    /** Lectura de la fecha **/
```

```
    printf("Introduzca el día: ");
```

```
    scanf("%d", &di);
```

```
    printf("Introduzca el mes: ");
```

```
    scanf("%d", &mes);
```

```
    /** Mostrar la estación del año **/
```

```
    /** (considerando que la fecha es válida) **/
```

```
    if ((mes==3 && di>=20) || mes==4 || mes==5 || (mes==6 && di<21))
```

```
        printf("Primavera\n");
```

```
    else if ((mes==6 && di>=21) || mes==7 || mes==7 || (mes==8 && di<22))
```

```
        printf("Verano\n");
```

```
    else if ((mes==9 && di>=22) || mes==10 || mes==11 || (mes==12 && di<21))
```

```
        printf("Otoño\n");
```

```
    else if ((mes==12 && di>=21) || mes==1 || mes==2 || (mes==3 && di<20))
```

```
        printf("Invierno\n");
```

```
    else {
```

```
        printf("Hay un error en la fecha\n");
```

```
        printf("Está mal el mes\n");
```

```
    }
```

```
    return 0;
```

```
}
```