



Nombre y apellidos: \_\_\_\_\_

**Notas previas:**

- Escribe tu nombre y apellidos en esta hoja e inmediatamente en todas las que cojas, incluso las de sucio. El no hacerlo puede suponer tu expulsión.
- Responde de manera clara y ordenada. Utiliza una cara para cada respuesta. Puedes utilizar lápiz.
- “Recibe” es distinto de “lee del teclado”. “Devuelve” es distinto de “escribe” o “muestra en pantalla”.

- [1 punto]** Indica en un recuadro lo que escribirá el siguiente programa en lenguaje C, explicando el razonamiento y explicitando los cálculos:

```
#include <stdio.h>
void EscribeDerechos (int pat);

void main (void)
{
    EscribeDerechos (23);
    EscribeDerechos (48);
    EscribeDerechos (421);
    EscribeDerechos (124);
}

void EscribeDerechos (int pat)
{
    printf ("%c%c%c\n",
            pat & 0x4? 'r':'-',
            pat & 0x2? 'w':'-',
            pat & 0x1? 'x':'-');
}
```

**Notas:**

- El operador & realiza una operación AND bit a bit
- En C se puede expresar una constante en hexadecimal anteponiéndole 0x. Por ejemplo, podemos expresar el 10 como 0xA.
- El tiempo de resolución y la posibilidad de confusión se puede reducir drásticamente si sólo realizas los cálculos estrictamente necesarios

- [3 puntos]** Codifica un programa en lenguaje C que lea un número *natural positivo* **n** y dibuje en pantalla una **estrella** de orden **n**, siguiendo os siguientes modelos de ejecución para **n=3**, **n=4** y **n=5** (en tipo de letra diferente el dato introducido):

Introduce el orden: <b>3</b>	Introduce el orden: <b>4</b>	Introduce el orden: <b>5</b>
1---1---	1----1----	1-----1-----
2--333--	2---333---	2----333----
3-55555-	3--55555--	3---55555---
2--333--	4-777777-	4--777777--
1---1---	3--55555--	5-99999999-
	2---333---	4--777777--
	1----1----	3---55555---
		2----333----
		1-----1-----



3. [1 punto] Diseña el **diagrama de flujo de la cabecera**, incluyendo *nombre*, parámetros de *entrada y salida* y *valor devuelto*, y el **prototipo** en el lenguaje C para las siguientes especificaciones:

- Función `Polin` que recibe los coeficientes de una ecuación lineal ( $a$  y  $b$ ) y obtiene la raíz  $x$ . La función devuelve un 1 ó un 0 según sea una ecuación de primer grado o no sea una ecuación, respectivamente.
- Función `SonCosmi` que recibe dos números,  $n_1$  y  $n_2$ , y devuelve un booleano que dice si los números son *intercósmicos* en cuyo caso calculará el grado de *intercosmicidad* en el parámetro `grd` de tipo real.
- Función `Primi` para verificar combinaciones de euromillones que recibe en un vector `boletos` los números jugados y en otro vector `premiados` la combinación ganadora y devuelve la categoría del premio, que será un número del 1 al 12, 0 si no hay premio.
- Función `MuestraPremios` que recibe la categoría de un boleto premiado y un vector `premios` con los importes en euros (con precisión de céntimos) por categoría y escribe en pantalla el premio obtenido.

**Nota importante:** no es necesario diseñar ni codificar las funciones.

4. [3 puntos] Estamos buscando un número  $f$  de determinadas características y sabemos que es **impar** y pertenece a la **serie** formada a partir de dos números  $n_1$  y  $n_2$  naturales y cada término  $i$ -ésimo ( $1 \leq i \leq 100$ ) se calcula de la siguiente manera:

$$f_i = n_1 \cdot i + n_2 / i$$

Diseña el **diagrama de flujo** y **codifica** un programa C que pida los números  $n_1$  y  $n_2$  y muestre los candidatos a partir de  $i = 1$ , mostrando como mucho 10. Para cada posible solución muestra su ordinal y el valor de la  $i$ , como en el siguiente ejemplo:

Introduce primer número: <b>12</b> Introduce segundo número: <b>37</b> 1 i: 1, fi: 49 2 i: 4, fi: 57 3 i: 5, fi: 67 4 i: 7, fi: 89 5 i: 10, fi: 123 6 i: 11, fi: 135 7 i: 12, fi: 147 8 i: 19, fi: 229 9 i: 20, fi: 241 10 i: 21, fi: 253	Introduce primer número: <b>18</b> Introduce segundo número: <b>7</b> 1 i: 1, fi: 25 2 i: 2, fi: 39 3 i: 4, fi: 73 4 i: 5, fi: 91 5 i: 6, fi: 109 6 i: 7, fi: 127
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Obsérvese que:

$n_1$ : <b>12</b> $n_2$ : <b>37</b> $f_1 = 12 \cdot 1 + 37 / 1 = 12 + 37 = \mathbf{49}$ (ok) $f_2 = 12 \cdot 2 + 37 / 2 = 24 + 18 = 42$ (par)	$n_1$ : <b>18</b> $n_2$ : <b>7</b> $f_7 = 18 \cdot 7 + 7 / 7 = 126 + 1 = \mathbf{127}$ (ok) $f_8 = 18 \cdot 8 + 7 / 8 = 144 + 0 = 144$ (par) $f_9 = 18 \cdot 9 + 7 / 9 = 162 + 0 = 162$ (par)
--------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



5. [2 puntos] (Elige entre esta pregunta o la siguiente) Disponemos de las funciones:

DiaSis	Obtiene <b>día, mes y año</b> del reloj del sistema (de hoy)
DiaJul	Devuelve una <b>fecha</b> en formato numérico (juliano)
DiaGrg	Convierte de formato juliano a formato día-mes-año ( <b>gregoriano</b> )

Sus prototipos, especificados en el fichero “**fechas.h**”, son los siguientes:

```
void DiaSis (int *dd, int *mm, int *aa);  
long DiaJul (int dd, int mm, int aa);  
void DiaGrg (long jul, int *dd, int *mm, int *aa);
```

Supóngase asimismo que **alguien nos proporciona** también de una función:

HorLib	Devuelve el potencial de horas libres de una fecha
--------	----------------------------------------------------

Su prototipo, especificado en el fichero “**agenda.h**”, es el siguiente:

```
float HorLib (long jul);
```

HorLib accede a una agenda que permite saber el número de horas disponibles para trabajar en una fecha futura dada.

Queremos escribir un programa que dado una estimación en número de horas para un proyecto nuevo nos muestre el calendario de días trabajados a partir de hoy (excluido) y la fecha de finalización, de la siguiente manera (único dato de entrada: 120):

Introduce horas del proyecto: <b>120</b>
28/01/2008 - 8.00 horas (quedan 112.00)
29/01/2008 - 8.00 horas (quedan 104.00)
30/01/2008 - 8.00 horas (quedan 96.00)
31/01/2008 - 8.00 horas (quedan 88.00)
01/02/2008 - 5.50 horas (quedan 82.50)
04/02/2008 - 8.00 horas (quedan 74.50)
05/02/2008 - 8.00 horas (quedan 66.50)
06/02/2008 - 8.00 horas (quedan 58.50)
07/02/2008 - 8.00 horas (quedan 50.50)
08/02/2008 - 5.50 horas (quedan 45.00)
11/02/2008 - 8.00 horas (quedan 37.00)
12/02/2008 - 8.00 horas (quedan 29.00)
13/02/2008 - 8.00 horas (quedan 21.00)
14/02/2008 - 8.00 horas (quedan 13.00)
15/02/2008 - 5.50 horas (quedan 7.50)
18/02/2008 - 7.50 horas (finalizado)



Escuela Universitaria  
de Ingeniería  
Vitoria-Gasteiz

Ingeniaritzako  
Unibertsitate Eskola  
Vitoria-Gasteiz

## Fundamentos de Informática

Examen – 25 de Enero de 2008

### Electrónica



6. [2 puntos] (Elige entre esta pregunta o la anterior) Queremos desarrollar una serie de funciones para operar con números complejos en forma binómica codificados mediante **vectores de dos elementos**. Así, podemos expresar el número complejo  $4.2 - 3.7i$  mediante un vector:

4.2	-3.7
-----	------

Diseña el **diagrama de flujo de la cabecera** y **codifica** las siguientes **funciones completas**:

- a. Función que calcule el Módulo sabiendo que dado un número complejo en forma binómica  $z = x + iy$  su módulo  $|z|$  se calcula mediante la fórmula:

$$|z| = \sqrt{x^2 + y^2}$$

Para calcular la raíz cuadrada puede utilizarse la función `sqrt` de la librería matemática (cuyo prototipo se proporciona en `math.h`):

```
double sqrt (double x);
```

- b. Función calcule el Conjugado (en la misma variable), sabiendo que dado un número complejo en forma binómica  $z = x + iy$  su conjugado  $\bar{z}$  es:

$$\bar{z} = x - iy$$

- c. Función lea un número complejo, como en el siguiente ejemplo:

Introduce la parte real: <b>4.2</b>
Introduce la parte imaginaria: <b>-3.7</b>

- d. Función escriba un número complejo, cada componente con dos decimales, mostrando únicamente un signo:

4.20 - 3.70i
--------------