



Nombre y apellidos: _____ Fecha _____

- Escribe tu **nombre** y **apellidos** en esta hoja e inmediatamente en todas las suplementarias, incluso las de sucio. El no hacerlo puede suponer tu expulsión.
- Todos los alumnos implicados en una copia** de un ejercicio **tendrán una nota final de 0**. El alumno es responsable de velar por su examen. Es decir **tanto el que copia como el que se deja copiar (ya sea de manera activa o pasiva)** recibirán el mismo castigo sin que exista atenuante alguno.
- Puedes utilizar **lápiz, bolígrafo, goma de borrar y corrector**. No puedes tener un **móvil** encendido ni utilizar **calculadora**.
- “Recibe” es distinto de “lee”. “Devuelve” o “calcula” es distinto de “escribe” o “muestra en pantalla”.
- Sé fiel a los textos de los ejemplos proporcionados. No serlo penaliza.

- [1,5 puntos]** **Copia** la siguiente tabla y **rellena** los huecos con el **Valor** en base decimal resultante de **evaluar las expresiones** correspondientes en orden secuencial, así como el valor modificado de cada una de las **variables** implicadas dada la declaración, suponiendo que sus valores se van guardando de una evaluación a la siguiente. Indica las operaciones realizadas junto a la tabla cuando no las realices mentalmente. El signo ? representa un valor desconocido. Si no cambia el valor de una variable deja la casilla vacía.

```
int a, b, c;
```

| Orden | Expresión | Valor | a | b | c |
|-------|-------------|-------|---|---|---|
| 0 | | – | ? | ? | ? |
| 1 | a = '\0' | | | | |
| 2 | b = 1.97 | | | | |
| 3 | c = 012 | | | | |
| 4 | a++ | | | | |
| 5 | --a | | | | |
| 6 | b^c | | | | |
| 7 | a += b | | | | |
| 8 | c%a | | | | |
| 9 | 0x10 && 1 | | | | |
| 10 | 0x10 & 1 | | | | |
| 11 | ~1+1 | | | | |
| 12 | 010 3 | | | | |
| 13 | 0101 010 | | | | |
| 14 | !-1 | | | | |
| 15 | 0x58>>3 | | | | |

Notas recordatorias:

- Los operadores simples | y & actúan bit a bit El operador ~ **complementa a uno** un número, es decir, realiza **bit a bit** una operación lógica **Not** de un número entero
- Los números negativos utilizan representación con **complemento a dos**
- Un desplazamiento actúa como una multiplicación o división entera por dos manteniendo el signo.



2. [1 punto] Dadas las siguientes especificaciones de subprogramas diseña la **cabecera del diagrama de flujo** con el *nombre*, los parámetros de *entrada*, los parámetros de *salida* y el *valor devuelto* por la función (cuando los haya), así como el **prototipo** de la función C correspondiente¹. Presta especial atención a la elección de los tipos de los datos.

- Recibe un código numérico para una antigua divisa europea (por ej. la peseta) y una cantidad en esa divisa y devuelve la cantidad correspondiente en euros.
- Recibe un dividendo y un divisor enteros y calcula el cociente y el resto de la división entera.
- Recibe las fechas de nacimiento de una madre y su hija en formato juliano (un único valor numérico de tipo entero largo) y devuelve la fecha (en el mismo formato) en la que la edad de la madre fue, es o será el doble de la edad de la hija (el cumpleaños de una de las dos).
- Escribe en pantalla la fecha del sistema.

3. [3,5 puntos] Conocemos que el número π puede calcularse utilizando la siguiente fórmula de Leibniz cuando n tienda a ∞ :

$$\frac{\pi}{4} = \sum_{i=0}^n \frac{(-1)^i}{2i+1}$$

- Codifica una función `Signo` en lenguaje C que reciba el i (el exponente natural) y calcule $(-1)^i$. Resuélvela a base de verificar si este exponente i es par o impar.
- Codifica una función `CalPi` en lenguaje C que reciba como parámetro el número de iteraciones n de la fórmula de Leibniz y devuelva la aproximación correspondiente de π . Utiliza la función del apartado anterior. Observa que el cálculo es incorrecto si se utilizan divisiones enteras.
- Para poder probar la función del apartado anterior, codifica una función `AnalizaCal` que reciba como parámetro un número de iteraciones y una cantidad de pruebas, mostrando sucesivamente en pantalla el número de iteraciones y el valor de π obtenido mediante llamadas sucesivas a la función del apartado anterior el valor calculado, con 12 decimales. Estudia el ejemplo proporcionado al final.
- Codifica el programa principal que pida y lea las iteraciones de partida y la cantidad de pruebas y llame adecuadamente a la función del apartado c. Ejemplo:

| Entradas y salidas del ejemplo | Sección |
|--|---|
| Iteraciones iniciales: 1500000 Cantidad de pruebas: 6 | Programa principal |
| 1500000: 3.141593320256 1500001: 3.141591986924 1500002: 3.141593320255 1500003: 3.141591986925 1500004: 3.141593320254 1500005: 3.141591986926 | Función <code>AnalizaCal</code> (que llama cada vez a <code>CalPi</code>) |

¹ **Nota importante:** no es necesario diseñar ni codificar el cuerpo de las funciones en este ejercicio 2.



4. [3 puntos] Disponemos de las funciones:

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

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);
```

- Diseña** el diagrama de flujo y **codifica** una **función** en lenguaje C que reciba una fecha de nacimiento de una persona y una fecha de referencia (ambas en formato juliano) y devuelva la edad para esa fecha. Si la persona no había nacido en la fecha dada devolverá un 0.
- Codifica** un programa en lenguaje C que pida una fecha de nacimiento y calcule la edad para esa persona en la fecha del sistema, utilizando la función del apartado anterior, escribiendo el resultado en pantalla. Ejemplo de ejecución:

Entradas y salidas del ejemplo

```
Introduce la fecha dd-mm-aaaa de tu nacimiento: 5-10-1975  
Tu edad hoy: 33
```

5. [1 punto] Escribe una
- función**
- que reciba una
- cadena de caracteres**
- como parámetro y devuelva la suma de los códigos ASCII de sus caracteres en un número entero.

Por ejemplo, como el carácter ‘2’ tiene el código ASCII 50 y el carácter ‘A’ tiene el código ASCII 65, para la cadena “222A222” nos devolverá 365.