

Fundamentos de Informática.

Nombre: _____

Titulación: _____

1. Contestar las siguientes preguntas. Cada respuesta acertada suma 0,5 puntos a la nota total y cada respuesta mal contestada penaliza con -0,3 puntos dicha nota. En cada pregunta sólo hay una única respuesta correcta (3p).

Un kilobyte (KB) equivale a

- medio megabyte (MB).
 - 1024 bytes.
 - 1024 MB.
 - 1000 bytes.
- Su tamaño depende del tipo concreto de memoria RAM que se esté usando.

La memoria principal es de acceso

- secuencial.
 - directo.
 - volátil.
- de un tipo de acceso propio de este tipo de memoria.

Un disco duro tiene

- tantos cilindros como pistas hay en cada superficie.
- tantos sectores como pistas hay en el disco.
- tantas pistas como cilindros hay en el disco.
- tantas pistas como superficies tiene el disco

Los monitores con tecnología LCD poseen una pantalla de cristal líquido, la cual es muy utilizada en los ordenadores portátiles debido a su pequeño grosor.

- Verdadero.
- Falso.

Un componente fundamental de una impresora LÁSER es

- la célula fotoeléctrica.
- el arco voltaico.
- el tambor fotosensible.
- el diodo LED.
- el convertidor D/A (Digital/Analógico).
- Todos los anteriores.
- Ninguno de los anteriores.

El módem se encarga de modular y demodular señales para posibilitar la comunicación de datos digitales por las líneas analógicas ya existentes (como la Red Telefónica Conmutada).

- Verdadero.
- Falso.

2. Enumerar de forma breve y concisa (no más de media página) las distintas tareas que suele realizar un Sistema Operativo (1p).
3. Diseñar un subalgoritmo en pseudocódigo al cual se le pase como único parámetro una matriz de enteros (`int`) de orden 5x5. El subalgoritmo deberá alterar el orden de los elementos de la matriz original para obtener su traspuesta. También se pide la codificación del subalgoritmo en lenguaje C (2p).
4. En lenguaje C existen cuatro funciones básicas para el manejo de ficheros de texto:

```
punt_fich = fopen(nombre_fichero, operación)
```

Para poder trabajar con un fichero antes debemos abrirlo. Para ello usaremos la función anterior, que acepta como primer parámetro el nombre del fichero en formato Ms-Dos (unidad, ruta, nombre del fichero) y como segundo parámetro una cadena de caracteres entre comillas que indica la operación a realizar ("r" para lectura y "w" para escritura). Si todo ha ido bien la función devuelve un puntero al fichero, que se almacenará en una variable de tipo `*FILE` (definido en el fichero `<stdio.h>`). Por el contrario, si hay algún error la función devolverá como resultado el valor cero. Cuando se usa esta función para abrir en escritura un fichero previamente inexistente, la propia función lo crea.

Para leer un carácter del fichero usaremos la siguiente función:

```
carácter = fgetc(punt_fich);
```

La variable *carácter* debe ser de tipo `signed char` y almacenará el carácter leído. Asimismo, la próxima vez que usemos la función leeremos el siguiente carácter del fichero. Si la última vez que la usamos leímos el último carácter del fichero, la siguiente vez leeremos un carácter especial: El carácter EOF (*fin de fichero*), el cual también se define en el fichero `<stdio.h>`.

Para escribir un carácter en el fichero usaremos la función:

```
fputc (carácter, punt_fich);
```

Finalmente, una vez que hayamos terminado de trabajar con el fichero deberemos cerrarlo, para lo cual usaremos la función `fclose (punt_fich);`

Se pide escribir un algoritmo en pseudocódigo que copie el contenido de un fichero de texto denominado "C:\PRUEBA\HOLA.TXT" en otro de nombre "C:\PRUEBA\ADIOS.TXT". También se pide la codificación del algoritmo en lenguaje C (4p).

Nota: Supondremos que no se produce ningún tipo de error en la apertura de ambos ficheros.

Fundamentos de Informática.

Primera Parte

1. Contestar las siguientes preguntas tipo test. Cada pregunta correctamente respondida suma 1 punto a la nota final, y cada pregunta erróneamente contestada penaliza con -0,8 puntos la nota final (5 p).

El Registro de Instrucción de la Unidad de Control contiene

- el número de identificación de la BIOS del ordenador.
- el código de operación del conjunto de instrucciones del programa.
- el código de operación de la instrucción a ejecutar.
- la instrucción a ejecutar escrita en un lenguaje de programación de alto nivel.
- Ninguna de las respuestas anteriores es correcta.

Relacionar los siguientes términos:

1 kilobytes	4 velocidad de transmisión
2 nanosegundos	5 frecuencia de reloj
3 caracteres por segundo	3 velocidad de impresión
4 bits por segundo	6 capacidad de proceso
5 megaherzios	1 capacidad de memoria
6 instrucciones por segundo	2 tiempo de acceso

En un sistema operativo, la redirección se utiliza para comunicar dos procesos.

- Verdadero.
- Falso.

En un sistema operativo tipo UNIX el *uid* es un número que identifica a cada usuario concreto.

- Verdadero.
- Falso.

Si queremos que un determinado programa se ejecute rápidamente será mejor

- programarlo utilizando un lenguaje de alto nivel.
- interpretarlo con un intérprete.
- compilarlo con un compilador.
- ejecutarlo bajo un sistema operativo.
- Ninguna de las respuestas anteriores es correcta.

2. Describir brevemente (en una página) las dos funciones básicas de un Sistema Operativo (2 p).
3. Explicar como funcionarían, en caso de que fuesen correctas, las siguientes órdenes en Ms-Dos. Contestar de forma *breve pero completa y razonada* (3 p):
- a) `C:\SYSTEM\DOS>type ..\unix.doc | prn > com1`
 - b) `C:\DOS>copy c:\docum*.* a:`
 - c) `C:\>dir \windows | sort > con`

Segunda Parte

1. Contestar las siguientes preguntas tipo test. Cada pregunta correctamente respondida suma 0,6 puntos a la nota final, y cada pregunta erróneamente contestada penaliza con -0,4 puntos la nota final (3 p).

El teorema de Bohm-Jacopini se basa en

- diseñar el algoritmo de modo general para después, mediante refinamientos sucesivos, ir aumentando progresivamente el nivel de detalle del diseño.
- escribir un algoritmo o un programa utilizando exclusivamente las estructuras de control secuencial, alternativa e iterativa.
- estructurar el programa en módulos de forma que éste resulte más fácil de entender y modificar.
- Ninguna de las respuestas anteriores es correcta.

Una variable es

- el identificador de un valor constante en un programa.
- un dato desconocido en un programa.
- un dato que posee un valor no constante y es conocido en un programa o un algoritmo por un identificador.
- Ninguna de las respuestas anteriores es correcta.

Las variables globales

- es preferible usarlas sólo cuando sea necesario, ya que si se utilizan resulta más difícil controlar que sus valores no se modifiquen erróneamente.
- se deben usar siempre que se pueda, ya que así se evita el paso de parámetros entre funciones.
- no se pueden usar en un programa escrito en lenguaje C, ya que crean conflictos irresolubles con las variables locales del programa.
- Ninguna de las respuestas anteriores es correcta.

En un programa en lenguaje C, el resultado de la operación $5. / 2$ es

- 2.0
- $25 * 10^{-2}$
- 2
- 2.5
- Ninguna de las respuestas anteriores es correcta.

Los modificadores de los tipos de datos básicos en lenguaje C son

- int y char.
- int, char, float y double.
- int, char, float, double y void.
- unsigned, signed, long y short.
- auto, extern, static y register.

2. Necesitamos diseñar cuatro funciones para obtener ciertos parámetros estadísticos a partir de una muestra. Los valores de la muestra están almacenados en un array unidimensional de tipo `double` y tamaño n . Todas las funciones aceptan como parámetros el array de datos y el tamaño de la muestra:

- a) Función `double med_arit(double v[], int n)`. Devuelve como resultado la media aritmética de los valores de la muestra (2p).
- b) Función `double med_geom(double v[], int n)`. Devuelve como resultado la media geométrica de los valores de la muestra (2p).
- c) Función `double varianza(double v[], int n)`. Devuelve como resultado la varianza de los valores de la muestra (2p).
- d) Función `double desv_tip(double v[], int n)`. Devuelve como resultado la desviación típica de los valores de la muestra (1p).

Notas:

- En todos los casos se piden los subalgoritmos escritos en pseudocódigo y la codificación de los mismos en lenguaje C.
- En el diseño de cada función pueden utilizarse cualquiera de las otras funciones con objeto de facilitar el trabajo.
- Para obtener la potencia n -sima de un dato real puede utilizarse la siguiente función, cuyo prototipo se encuentra en `<math.h>`:

```
double pow(double base, double exponente)
```

Fundamentos de Informática.

Primera Parte

1. Contestar las siguientes preguntas tipo test. Cada pregunta correctamente respondida suma 1 punto a la nota final, y cada pregunta erróneamente contestada penaliza con -0,8 puntos la nota final (5 p).

La memoria ROM sirve para almacenar

- información que nunca se va a modificar.
- datos e instrucciones de forma temporal.
- información que necesitemos almacenar por poco tiempo.
- Ninguna de las respuestas anteriores es correcta.

La memoria principal es de acceso

- secuencial.
- volátil.
- directo.
- de un tipo de acceso propio de este tipo de memoria.

En un sistema operativo tipo UNIX el *pid* es

- un número que identifica al sistema de ficheros concreto que se está utilizando.
- un número que indica la prioridad de cada proceso.
- un número que indica el tiempo estimado de ejecución de cada proceso.
- un número que identifica a cada proceso.
- Ninguna de las respuestas anteriores es cierta.

En un sistema operativo, la tubería se utiliza para comunicar dos procesos.

- Verdadero.
- Falso.

La diferencia fundamental entre un compilador y un intérprete es

- que el intérprete genera código objeto entendible por el microprocesador.
- que el compilador genera código objeto entendible por el microprocesador.
- que el compilador a usar depende de la BIOS del ordenador concreto.
- no hay diferencias ya que ambos son traductores.
- Todas las respuestas anteriores son ciertas.

2. Se dispone de un monitor de 640x480 píxeles de resolución, con 256 colores posibles y 0.28 mm de separación entre píxeles. Este monitor está conectado a un ordenador cuyo microprocesador trabaja a una frecuencia de reloj de 200 Mhz. Además, el tiempo que tarda el monitor en dibujar cada píxel es 25 ns. (nanosegundos). Hallar razonadamente (3 p):

- a) Frecuencia de barrido vertical del monitor.
- b) Frecuencia de barrido horizontal.
- c) Cantidad mínima de memoria de vídeo expresada en kilobytes necesaria para representar una imagen en la resolución indicada.

3. Supongamos que tenemos una tarjeta de sonido que puede digitalizar sonidos o música con la calidad de un Compact Disk de audio normal (CD-A), con 44,1 KHz. de frecuencia de muestreo y 12 bits de tamaño de muestra. ¿Qué cantidad de memoria (en megabytes) necesitaríamos para almacenar la canción de *Mecano* titulada “Los piratas del amor” de su disco “El club de los humildes”, que dura cinco minutos y tres segundos. Razónese la respuesta (2 p).

Segunda Parte

1. Contestar las siguientes preguntas tipo test. Cada pregunta correctamente respondida suma 0,6 puntos a la nota final, y cada pregunta erróneamente contestada penaliza con -0,4 puntos la nota final (3 p).

Un algoritmo es

- un término equivalente en significado al hardware.
- un conjunto finito y ordenado de reglas que permite resolver un problema.
- la codificación de un programa.
- un lenguaje de programación estructurado.
- Ninguna de las respuestas anteriores es correcta.

En la Programación Estructurada las estructuras de control son

- secuencial, condicional y declarativa.
- secuencial, condicional y bifurcativa.
- exponencial, condicional y recursiva.
- No existen, ya que los programas codificados en un lenguaje estructurado usan la sentencia `goto` para realizar los saltos condicionales e incondicionales.
- Ninguna de las respuestas anteriores es correcta.

El símbolo `&` en el lenguaje C

- se utiliza para modificar el valor de una constante.
- es el operador de indirección.
- es el operador de dirección.
- indica el valor de retorno de una función.
- Ninguna de las respuestas anteriores es correcta.

Si en el cuerpo de un bucle `while (test) { ... }` se usa la sentencia `continue;`

- se ejecutará la siguiente sentencia dentro del cuerpo del bucle.
- se ejecutará la última sentencia que haya dentro del cuerpo del bucle.
- se evaluará la expresión de `test` para continuar o no ejecutando el bucle.
- se ejecutará de nuevo el bucle independientemente del valor de la expresión de `test`.
- Ninguna de las respuestas anteriores es correcta.

En un programa escrito en lenguaje C

- los arrays se pasan por valor.
- se pasa por valor la dirección de memoria del primer elemento del array.
- no existe ninguna manera de pasar un array como parámetro a una función.
- sólo pueden declararse arrays que sean globales a todas las funciones del programa.
- Ninguna de las respuestas anteriores es correcta.

2. Necesitamos diseñar cuatro funciones que realicen determinadas operaciones con números complejos. Cada número complejo se representará como un array unidimensional de tamaño 2 y de tipo double. Las funciones son las siguientes:

- a) Función `void cong(double a[])`. Transforma el número complejo `a` en su conjugado (1 p).
- b) Función `void sum_rest(double a[], double b[], double c[], char op)`. Opera los complejos `a` y `b`, y almacena el resultado en `c`. La operación será una suma si el parámetro `op` vale cero o una resta si dicho parámetro vale uno (2 p).
- c) Función `double modulo(double a[])`. Devuelve el módulo del complejo `a` (2 p).
- d) Función `void binomica(double mod, double arg, double a[])`. Obtiene la expresión binómica de un complejo representado en forma polar (2 p).

Notas:

- En todos los casos se piden los subalgoritmos escritos en pseudocódigo y la codificación de los mismos en lenguaje C.
- Para obtener la raíz cuadrada de un número real, el seno y el coseno de un ángulo pueden utilizarse respectivamente las siguientes funciones, cuyos prototipos se encuentran en el fichero de encabezamiento `<math.h>`:

```
double sqrt(double numero)
double sin(double angulo)
double cos(double angulo)
```

Fundamentos de Informática.

1. Contestar las siguientes preguntas tipo test. Cada pregunta correctamente respondida suma 0,7 puntos a la nota final, y cada pregunta erróneamente contestada penaliza con -0,5 puntos la nota final (3,5 p).

Los sistemas operativos

- planifican y supervisan la ejecución de los programas.
- controlan las operaciones de entrada y salida de datos.
- administran los recursos del ordenador.
- Todas las respuestas anteriores son correctas.
- Ninguna de las respuestas anteriores es correcta.

Desde el *prompt* del sistema operativo MS-DOS se pueden ejecutar los ficheros con extensión

- .EXE.
- .EXE y .COM.
- .EXE y .BAT.
- .EXE, .COM y .BAT.
- .EXE, .COM, .BAT y .SYS.
- Ninguno de los anteriores.

El teorema de Bohm-Jacopini se basa en

- diseñar el algoritmo de modo general para después, mediante refinamientos sucesivos, ir aumentando progresivamente el nivel de detalle del diseño.
- utilizar sólo tipos de datos simples: entero, real, carácter y lógico.
- escribir un algoritmo o un programa utilizando exclusivamente las estructuras de control secuencial, alternativa e iterativa.
- Todas las respuestas anteriores.

¿Cuál es la salida del siguiente programa?

```
#include <stdio.h>

int a = 1;

void main() {
    int a = 0;

    printf("\nLa variable \"a\" vale: %d.", a);
    return;
}
```

- Uno, porque las variables globales siempre tienen preferencia ante las locales.
- Cero, porque al ser la variable *a* local a la función *main()*, dentro de dicha función la declaración local prevalece sobre la global.
- Se trata de un error, ya que no pueden existir dos variables con el mismo identificador aunque tengan distinto ámbito.
- Ninguna de las respuestas anteriores es correcta.

El operador de redirección "<" redirige la salida estándar de un proceso a un fichero cuyo nombre se debe indicar a la derecha del operador.

- Verdadero.
- Falso.

2. El N.I.F. (Número de Identificación Fiscal) es exclusivo para cada persona y está formado por el número del D.N.I. (Documento Nacional de Identidad) y una letra añadida, la cual depende directamente del D.N.I. y cuya utilidad consiste en permitir la detección de posibles errores al introducir un determinado D.N.I. Es simplemente un dígito de control de errores.

El algoritmo para obtener esta letra a partir del D.N.I. es el siguiente: Se calcula el resto de dividir el D.N.I. entre 23. Dicho resto está comprendido entre 0 y 22. Entonces, a cada uno de esos posibles restos se le asigna respectivamente una de las siguientes letras: T, R, W, A, G, M, Y, U, P, D, X, B, N, J, Z, S, Q, V, H, L, C, K y E. Se ha escogido el número 23 por ser el número primo más grande que es menor que el número de letras en el abecedario. En español las letras son 28 y el siguiente número primo después del 23 es el 29. Además hay letras que no pueden asignarse por estar formadas por dos caracteres.

Diseñar un algoritmo mediante pseudocódigo que pida por teclado un número de D.N.I. y posteriormente imprima la letra del N.I.F. que le corresponde. También se pide la codificación del algoritmo en lenguaje C para obtener el programa correspondiente. Téngase en cuenta que el D.N.I. es un número muy grande y no puede almacenarse en un dato de tipo `int` (1,5 p).

3. Necesitamos diseñar cuatro funciones que realicen determinadas operaciones con números complejos. Cada número complejo se representará como un array unidimensional de tamaño 2 y de tipo `double`. Las funciones son las siguientes:

- a) Función `void escalar(double a[], double num)`. Multiplica el número complejo `a` por el escalar `num` (1 p).
- b) Función `void suma(double a[], double b[], double c[])`. Suma los complejos `a` y `b`, y almacena el resultado en `c` (1 p).
- c) Función `double modulo(double a[])`. Devuelve el módulo del complejo `a` (1 p).
- d) Función `void producto(double a[], double b[], double c[])`. Multiplica los complejos `a` y `b`, y almacena el resultado en `c` (2 p).

Notas:

- En todos los casos se piden los subalgoritmos escritos en pseudocódigo y la codificación de los mismos en lenguaje C.
- Para obtener la raíz cuadrada de un número real puede utilizarse la siguiente función, cuyo prototipo se encuentra en el fichero `<math.h>`:

```
double sqrt(double numero)
```

Fundamentos de Informática.

1. Contestar las siguientes preguntas tipo test. Cada pregunta correctamente respondida suma 0,8 puntos a la nota final, y cada pregunta erróneamente contestada penaliza con -0,6 puntos la nota final (4 p).

Una dirección de memoria es

- un conjunto aleatorio de bits.
- un conjunto de bits que indica si los datos están en memoria principal o en disco duro.
- un conjunto de bits que indica la posición concreta de una celda de memoria.
- la localización de la memoria en el microprocesador.
- Ninguna de las respuestas anteriores es cierta.

Los sistemas operativos

- planifican y supervisan la ejecución de los programas.
- controlan las operaciones de entrada y salida de datos.
- administran los recursos del ordenador.
- Todas las respuestas anteriores son correctas.
- Ninguna de las respuestas anteriores es correcta.

En un sistema operativo un *proceso* es

- el código ejecutable de un programa.
- cualquiera de las funciones básicas del sistema operativo.
- un sistema de gestión de ficheros.
- un programa en ejecución.
- Ninguna de las respuestas anteriores es correcta.

¿Cuál es la salida del siguiente programa?

```
#include <stdio.h>

int a = 0;

void main() {

    int a = 1;

    printf("\nLa variable \"a\" vale: %d.", a);
    return;
}
```

- Se trata de un error, ya que no pueden existir dos variables con el mismo identificador aunque tengan distinto ámbito.
- Cero, porque las variables globales siempre tienen preferencia ante las locales.
- Uno, porque al ser la variable *a* local a la función *main()*, dentro de dicha función la declaración local prevalece sobre la global.
- Ninguna de las respuestas anteriores es correcta.

El símbolo *** en el lenguaje C

- se usa para modificar el valor de una constante.
- es el operador de indirección.
- es el operador de dirección.
- indica el valor de retorno de una función.
- Ninguna de las respuestas anteriores es correcta.

2. Definición de algoritmo. Propiedades (2 p).

3. Codificar en lenguaje C un programa que calcule numéricamente la integral de la función de onda $o(x)=a \cos(kx + p)$ entre los límites de integración g y h . Dicha integral es el área del polígono limitado por el eje de abscisas, los límites de integración y la propia curva de la función. Por lo tanto, el programa deberá pedir en primer lugar los coeficientes de la función (a , k y p) y los límites de integración (g y h).

Este área se calculará aproximadamente, y para ello se pedirá como entrada el número de particiones, las cuales dividirán al polígono anterior en subpolígonos. Éstos se aproximarán a trapecios y el resultado que buscamos será la suma de sus áreas. Cada trapecio tendrá como base el tamaño de la partición y como alturas los valores absolutos de la función en los dos extremos de la base. Cada área se calculará como el producto de la semisuma de las dos alturas por la base (4 p).

Notas: También se pide el pseudocódigo, y *se puntuará negativamente una modularización incorrecta de éste y del programa resultante.*

Para el desarrollo del ejercicio se utilizarán las funciones $\text{abs}()$ y $\text{cos}()$, cuyos prototipos se encuentran en el fichero de encabezamiento `<math.h>`.

Fundamentos de Informática.

1. Contestar las siguientes cinco cuestiones. Cada cuestión totalmente correcta suma 0,4 puntos a la nota total y cada cuestión fallida penaliza con -0,3 puntos dicha nota (2p).

Indicar qué afirmaciones son correctas:

- La memoria RAM se usa para el almacenamiento masivo de información.
- La memoria ROM es de sólo lectura.
- La memoria EPROM se puede borrar mediante luz ultravioleta.
- Los soportes de almacenamiento masivo son más rápidos que la memoria RAM

La información de un disco duro se organiza en

- superficies, pistas y cilindros.
- pistas y sectores.
- superficies y sectores.
- Ninguna de las anteriores es correcta.

Ms-Dos es un sistema operativo multiprocesador y multitarea.

- Verdadero.
- Falso.

Indicar qué afirmaciones son correctas:

- La reutilización del software se consigue mediante el empleo de funciones.
- Las funciones suelen agruparse en bibliotecas que luego se pueden utilizar en cualquier programa.
- Mediante la programación modular (con funciones) se desarrolla un software menos legible.
- Un programa escrito en ensamblador es más fácil de modificar que escrito en C.

Indicar cuál de las siguientes no es una característica de un programa estructurado:

- Codifica un algoritmo bien diseñado y utiliza estructuras de control adecuadas.
- Se ha desarrollado mediante una metodología de diseño descendente.
- Utiliza la sentencia GO TO.
- Utiliza funciones, de modo que el programa adquiere una estructura modular.

2. Explicar de forma breve y concisa (no más de media página) las funciones básicas de un Sistema Operativo (2p).
3. Necesitamos escribir un programa con cuatro funciones para tratar estadísticamente los valores de una muestra. Éstos se almacenan en un array unidimensional de tipo `double` y tamaño n . Todas las funciones aceptan como parámetros el array de datos y el tamaño de la muestra.
- a) Función `void leer_muestra(double v[], int n)`. Lee por teclado los n valores de la muestra y los almacena según el orden de lectura (1p).

- b) Función `double med_arit(double v[], int n)`. Devuelve como resultado la media aritmética de los valores de la muestra (1,5p).
- c) Función `double varianza(double v[], int n)`. Devuelve como resultado la varianza de los valores de la muestra (1,5p).
- d) Función `void escr_tip(double v[], int n)`. Escribe por pantalla los `n` valores de la muestra en líneas separadas (1p).
- e) Un programa principal que utilice las funciones anteriores para almacenar en un vector una muestra de veinte datos e imprimir a continuación dichos datos, la media aritmética y la varianza (1p).

Notas:

- Sólo se pide la codificación del programa principal y de las funciones en lenguaje C.
- En el diseño de cada función pueden utilizarse cualesquiera de las otras funciones con objeto de facilitar el trabajo.

Fundamentos de Informática.

Primera Parte

Cada pregunta correctamente contestada sumará 1 punto a la calificación total. En caso de error se penalizará ésta con -0,5 puntos.

1. Un bit es
 - la unidad de medida de la velocidad de transmisión de información.
 - la unidad mínima utilizada para medir la información.
 - el número de bytes necesarios para almacenar un carácter.
 - ocho bytes.

2. El término "informática" significa *literalmente* "tratamiento de la información".
 - Verdadero.
 - Falso.

3. La memoria RAM se utiliza para almacenar
 - información que necesitamos almacenar por mucho tiempo.
 - datos e instrucciones de forma temporal.
 - el programa que se encarga de realizar los diagnósticos del ordenador cuando éste es encendido.
 - nada de lo anterior.

4. La CPU es
 - la Unidad Central de Periféricos.
 - el software encargado de procesar la información.
 - el proceso asignado a un sistema operativo de disco.
 - el dispositivo electrónico encargado de procesar la información.

5. Un programa que desarrollemos en lenguaje máquina para un procesador puede ser ejecutado en cualquier otro de distinta arquitectura sin tener que modificarlo previamente.
 - Verdadero.
 - Falso.

6. La información de un disco duro se organiza en
 - pistas y cilindros.
 - superficies, pistas y sectores.
 - superficies, pistas y cilindros.
 - pistas, sectores y unidades de asignación.

7. El módem se encarga de modular y demodular señales para posibilitar la comunicación de datos digitales por las líneas analógicas ya existentes.
 - Verdadero.
 - Falso.

8. En un S.O. tipo UNIX, el *uid* nos indica el tiempo estimado de ejecución para cada proceso.
 - Verdadero.
 - Falso.

9. En MS-DOS las órdenes externas permanecen siempre en la memoria principal del ordenador.
 - Verdadero.
 - Falso.

10. En MS-DOS los ficheros tienen una longitud máx. para el nombre y la extensión de 10 y 3 caracteres respectivamente.
 - Verdadero.
 - Falso.

Fundamentos de Informática.

Segunda Parte

1. Dado la siguiente función en lenguaje C:

```
int misterio (int num) {  
  
    int a = num, b = 0;  
    int c;  
  
    while (a > 0) {  
        c = a % 10;  
        b = b * 10 + c;  
        a /= 10; /* división entera */  
    }  
  
    if (b == num) return 1;  
    else return 0;  
}
```

Se pide realizar el juego de ensayo de la misma para los siguientes valores: 174, 1225, 7890, 21412 y 8. En todo momento debe indicarse el valor que tomen las distintas variables de la función, especialmente en cada una de las iteraciones que se efectúen. A la vista de los resultados, ¿Podría indicar cuál es el problema que resuelve la función? (2p)

2. Necesitamos diseñar cinco funciones que realicen determinadas operaciones con números complejos. Cada número complejo se representará como un array unidimensional de tamaño 2 y de tipo double. Las funciones son las siguientes:

- Función void `escalar(double a[], double num)`. Multiplica el número complejo a por el escalar num (1,5 p).
- Función void `suma(double a[], double b[], double c[])`. Suma los complejos a y b, y almacena el resultado en c (1,5 p).
- Función double `modulo(double a[])`. Devuelve el módulo del complejo a (1,5 p).
- Función void `producto(double a[], double b[], double c[])`. Multiplica los complejos a y b, y almacena el resultado en c (2 p).
- Función void `vector(double mod, double arg, double a[])`. Devuelve las componentes vectoriales del complejo expresado en forma polar mediante el módulo mod y el argumento arg (1,5 p).

Nota:

- Para obtener la raíz cuadrada de un número real y el seno y el coseno de un ángulo pueden utilizarse respectivamente las siguientes funciones, cuyos prototipos se encuentran en el fichero de encabezamiento `<math.h>`:

```
double sqrt(double numero)  
double sin(double angulo)  
double cos(double angulo)
```

Fundamentos de Informática.

Primera Parte

Cada pregunta correctamente contestada sumará 1 punto a la calificación total. En caso de error se penalizará ésta con -0,5 puntos.

1. El término "informática" significa literalmente "tratamiento automático de la información".
 Verdadero.
 Falso.
2. El módem se encarga de modular y demodular señales para posibilitar la comunicación de datos analógicos por las líneas digitales ya existentes.
 Verdadero.
 Falso.
3. El componente principal de una impresora láser es el tambor fotosensible.
 Verdadero.
 Falso.
4. Los monitores con tecnología CRT poseen una pantalla de cristal líquido o cuarzo líquido, la cual es muy utilizada en los ordenadores portátiles por su pequeño grosor.
 Verdadero.
 Falso.
5. Indicar cuáles de las siguientes afirmaciones son correctas:
 Los sistemas operativos planifican y supervisan la ejecución de los programas.
 Los sistemas operativos administran los recursos del ordenador.
 Los sistemas operativos traducen los programas escritos en lenguaje máquina a un lenguaje de alto nivel.
 Ninguna de las afirmaciones anteriores es correcta.
6. En un S.O. un *proceso* es
 el sistema de ficheros.
 el planificador de tareas.
 un programa en ejecución.
 Ninguna de las respuestas anteriores.
7. En un S.O. tipo UNIX, el *pid* nos indica el tiempo estimado de ejecución para cada proceso.
 Verdadero.
 Falso.
8. En MS-DOS las órdenes externas permanecen siempre en la memoria principal del ordenador.
 Verdadero.
 Falso.
9. ¿En qué mejora Windows 3.11 a MS-DOS?
 En la utilización de la multitarea.
 En el uso de un entorno gráfico amigable y común a las aplicaciones.
 En que permite utilizar toda la memoria RAM disponible en el ordenador.
 En nada de lo anterior.
10. Indicar cuáles de las siguientes afirmaciones son correctas:
 Las instrucciones del lenguaje ensamblador son de tipo mnemotécnico o simbólico.
 Los programas escritos en lenguajes de alto nivel son poco transportables entre ordenadores.
 En un programa en lenguaje máquina pueden incluirse comentarios para facilitar su legibilidad.
 Ninguna de las afirmaciones anteriores es correcta.

Fundamentos de Informática.

Segunda Parte

1. El N.I.F. (Número de Identificación Fiscal) es exclusivo para cada persona y está formado por el número del D.N.I. (Documento Nacional de Identidad) y una letra añadida, la cual depende directamente del D.N.I. y cuya utilidad consiste en permitir la detección de posibles errores al introducir un determinado D.N.I. Es simplemente un dígito de control de errores.

El algoritmo para obtener esta letra a partir del D.N.I. es el siguiente: Se calcula el resto de dividir el D.N.I. entre 23. Dicho resto está comprendido entre 0 y 22. Entonces, a cada uno de esos posibles restos se le asigna respectivamente una de las siguientes letras: T, R, W, A, G, M, Y, U, P, D, X, B, N, J, Z, S, Q, V, H, L, C, K y E. Se ha escogido el número 23 por ser el número primo más grande que es menor que el número de letras en el abecedario. En español las letras son 28 y el siguiente número primo después del 23 es el 29. Además hay letras que no pueden asignarse por estar formadas por dos caracteres.

Codificar una función a la cual se le pase como parámetro un número de D.N.I. y suministre como valor de retorno la letra del N.I.F. que le corresponde. Téngase en cuenta que el D.N.I. es un número muy grande y no puede almacenarse en un dato de tipo `int` (2 p).

2. Necesitamos escribir un programa con cuatro funciones para tratar estadísticamente los valores de una muestra. Éstos se almacenan en un array unidimensional de tipo `double` y tamaño `n`, el cual es local al programa principal. Todas las funciones, excepto la función principal, aceptan como parámetros el vector de datos y el tamaño de la muestra. Se pide codificar un programa en lenguaje C conforme a los siguientes apartados:

- a) Función `void leer_muestra(double v[], int n)`. Lee por teclado los `n` valores de la muestra y los almacena en el array según el orden de lectura (1p).
- b) Función `double med_arit(double v[], int n)`. Devuelve como resultado la media aritmética de los valores de la muestra (2p).
- c) Función `double varianza(double v[], int n)`. Devuelve como resultado la varianza de los valores de la muestra (2p).
- d) Función `void escr_muestra(double v[], int n)`. Escribe por pantalla los `n` valores de la muestra en líneas separadas (1p).
- e) Programa principal `void main()`. Utiliza las funciones anteriores para almacenar una muestra de *veinte datos* e imprimir a continuación dichos datos, la media aritmética y la varianza (2p).

Nota:

- En el diseño de cada función pueden utilizarse cualesquiera de las otras funciones con objeto de facilitar el trabajo.

Fundamentos de Informática.

1. Responder las siguientes preguntas teniendo en cuenta que cada una sumará 0,4 puntos a la nota final si se contesta correctamente, y *restará* 0,3 en caso de error (2p):

La CPU es

- la Unidad Central de control de Periféricos.
- la parte del sistema operativo que controla el hardware del ordenador.
- el hardware encargado de procesar la información.
- el proceso asignado a un sistema operativo de disco (DOS).
- Todas las respuestas anteriores son ciertas.

¿Por cuál de los puertos circulan los bits secuencialmente a través de una única línea de transmisión?

- Paralelo.
- Serie.
- Bus PCI.
- Bus AGP.
- Interfaz IDE.
- Ninguna de las respuestas anteriores es cierta.

En MS-DOS los comandos externos

- permanecen siempre en memoria principal.
- están almacenados en ROM y se ejecutan cuando son requeridos.
- forman parte de la BIOS.
- están almacenados en ficheros.
- no existen, sólo existen los comandos internos.

Desde el *prompt* del sistema operativo MS-DOS se pueden ejecutar los ficheros con extensión

- .EXE.
- .EXE y .COM.
- .EXE y .BAT.
- .EXE, .COM y .BAT.
- .EXE, .COM, .BAT y .SYS.
- Ninguno de los anteriores.

El teorema de Bohm-Jacopini se basa en

- diseñar el algoritmo de modo general para después, mediante refinamientos sucesivos, ir aumentando progresivamente el nivel de detalle del diseño.
- utilizar sólo tipos de datos simples: entero, real, carácter y lógico.
- escribir un algoritmo o un programa utilizando las estructuras de control secuencial, condicional e iterativa (sin hacer uso de la sentencia *goto*).
- Todas las respuestas anteriores.

2. Codificar una función en lenguaje C que acepte como parámetro un número entero positivo y devuelva como resultado otro entero, que valdrá uno (1) si el número anterior es primo y cero (0) si no lo es. Además, la función deberá imprimir por pantalla todos los divisores de dicho número (2p).

Ejemplo: El número 16 es compuesto (el resultado de la función valdrá cero) y sus divisores son 16, 8, 4, 2 y 1.

3. Se va a escribir un programa que opera con matrices cuadradas de orden 3x3. Cada elemento de la matriz es un número real de tipo `double`. Para soportar la estructura matricial deberá utilizarse *obligatoriamente* el dato estructurado `array`. Las operaciones a realizar son:

- a) Suma y resta de matrices (1,5p).
- b) Determinar si una matriz es simétrica (1,5p).
- c) Producto de matrices (1,5p).
- d) Producto de un escalar por una matriz (1,5p).

Se pide la codificación en C de cuatro funciones, con su cabecera y cuerpo correspondiente, las cuales deben realizar respectivamente cada una de las operaciones anteriormente indicadas:

- a) Función `suma_resta()`: acepta como parámetros tres matrices, donde las dos primeras son los operandos y la tercera la matriz suma o diferencia, según el caso. También aceptará como cuarto parámetro un carácter que será el signo de la operación a realizar ('+' ó '-').
- b) Función `simétrica()`: acepta como parámetro una matriz y devuelve como resultado uno (1) si la matriz es simétrica y cero (0) en caso contrario.
- c) Función `producto()`: acepta como parámetros tres matrices, donde la última es el producto de las dos anteriores.
- d) Función `escalar()`: acepta como parámetros un escalar de tipo `double` y dos matrices, donde la segunda matriz es el resultado de multiplicar el escalar por la primera.

Fundamentos de Informática.

1. Responder las siguientes preguntas teniendo en cuenta que cada una sumará 0,4 puntos a la nota final si se contesta correctamente, y *restará* 0,3 en caso de error (2p):

El término “informática” significa literalmente “tratamiento automático de la información”.

- x Verdadero.
 Falso.

Un kilobyte (KB) equivale a

- medio megabyte (MB).
 x 1024 bytes.
 1024 MB.
 1000 bytes.
 Su tamaño depende del tipo concreto de memoria RAM que se esté utilizando.

La memoria principal es de acceso

- secuencial.
 volátil.
 magnético.
 x directo.
 de un tipo de acceso propio de esta clase de memoria.

Indicar cuáles de los siguientes dispositivos son apuntadores:

- Plotter.
 Scanner.
 Ratón.
 x Trackball.
 Impresora.
 Joystick.
 Tarjeta de sonido.

MS-DOS es

- un sistema operativo potente, multiusuario y multitarea.
 un sistema operativo tipo UNIX para estaciones de trabajo.
 un sistema operativo monousuario para ordenadores personales.
 un sistema operativo multiusuario con un potente interfaz de usuario gráfico.
 Ninguna de las respuestas anteriores es correcta.

2. Codificar en lenguaje C dos funciones para manipular cadenas de caracteres. La primera de ellas acepta como parámetro una cadena de caracteres y devuelve la longitud de la misma, excluyendo el carácter *fin de cadena* ('\0'). La segunda recibe como parámetros dos cadenas de caracteres, copia el contenido de la primera cadena en la segunda y añade el carácter de fin de cadena al final de ésta última. Además, suministra como valor de retorno la dirección de comienzo de la segunda cadena. Esta última función no tiene por qué comprobar si la segunda cadena posee suficiente espacio para almacenar los caracteres de la primera más el carácter fin de cadena. Los prototipos de ambas funciones son:

- a) `int cadlong(char []);` (1p)
- b) `char *cadcopia(char [], char []);` (1p)

3. Se va a escribir un programa en lenguaje C que opera con matrices cuadradas de orden 3x3. Cada elemento de la matriz es un numero real de tipo `double`. Para soportar la estructura matricial deberá utilizarse *obligatoriamente* el dato estructurado `array`. Las operaciones a realizar son:

- a) Inicialización (1,5p).
- b) Comparación (1,5p).
- c) Transposición (1,5p).
- d) Producto especial (1,5p).

Se pide la codificación en C de cuatro funciones, con su cabecera y cuerpo correspondiente, las cuales deben realizar respectivamente cada una de las operaciones anteriormente indicadas:

- a) Función `inicia()`: acepta como parámetros una matriz y un escalar de tipo `double`. Para inicializar la matriz se asocia a cada elemento de la misma un entero consecutivo empezando por la unidad, y a continuación se asigna a cada elemento el producto del entero correspondiente por el escalar. Los elementos de la matriz se enumeran por filas.
- b) Función `compara()`: acepta como parámetro dos matrices y devuelve como resultado uno (1) si ambas son iguales y cero (0) en caso contrario.
- c) Función `traspuesta()`: acepta como parámetros dos matrices, donde la segunda es la traspuesta de la primera.
- d) Función `especial()`: acepta como parámetros tres matrices, donde a cada elemento de la tercera matriz se le asigna el producto de los dos elementos que ocupan la misma posición en las matrices factores.

