

Universidad Simón Bolívar
Departamento de Computación y
Tecnología de la Información
CI-2125 Computación I
CI-2127 Algoritmos y Programación

Sección: _____ Carnet: _____

Nombre: _____

Tercer Parcial

25%

PARTE 1: (Valor 5 puntos)

Diseñe un algoritmo que realice la lectura consecutiva, por teclado, de números enteros hasta que se introduzca el valor 0 (cero) o se hayan introducido 20 números, y los almacene en un arreglo, en orden inverso al de su introducción. Luego se escribe el arreglo obtenido. Escriba las especificaciones entrada y salida, las pre y post condiciones, así como el diagrama de flujo.

ENTRADAS :

PRECONDICIÓN :

SALIDAS :

POSTCONDICIÓN :

DIAGRAMA DE FLUJO :

PARTE II: (12 puntos)

Se desea implementar un programa en C que maneje un arreglo de estructuras para almacenar las fechas de cumpleaños de 20 amigos. Los datos a almacenar sobre cada amigo son nombre, día, mes y año. El programa debe indicarnos cuántos de nuestros amigos cumplen los años un día y mes determinados introducidos por teclado. Complete el código de los siguientes apartados, suponiendo que se definen las siguientes estructuras para FECHA y AMIGO

```
typedef struct tipoFecha {
    int dia;
    int mes;
} FECHA;

typedef struct tipoAmigo {
    char nombre[30];
    char apellido[30];
    FECHA cumple;
} AMIGO;
```

1. (2 puntos) Definición de un subprograma que inicializa un arreglo de n amigos.

```
void Inicializar (AMIGO agenda[], int n)
//PRE: n > 0
//POST: inicializa el arreglo con los datos de n amigos
{
    int i;

}
}
```

2. (2 puntos) Definición de un subprograma que muestra los datos de un arreglo de n amigos.

```
void Mostrar (AMIGO agenda[], int n)
// PRE: n>0
// POST: Muestra los datos de n amigos
{
    int i;

}
}
```

3. (2 puntos) Definición de un subprograma que muestra en pantalla los datos de los amigos que cumplen años en una fecha dada.

```
void Buscar_Amigos_Fecha(AMIGO agenda[], int n, FECHA f)
//PRE: n>0
//POST: Escribe todos los amigos que cumplen años en f o un mensaje
//indicando que no encuentro ninguno.
{
    int i, contador;

}
}
```

4. (3 puntos) Definición de un subprograma que muestra en pantalla los datos de los amigos cuyo nombre o apellido es el dado.

```
void Buscar_Amigos_Nombre(AMIGO agenda[], int n, char s[])
//PRE: n>0 y s contiene un nombre o apellido
//POST: Escribe todos los amigos cuyo nombre o apellido es s, o un
//mensaje indicando que no encuentro ninguno.
{
    int i, contador;
}
}
```

5. (3 puntos) Programa principal que inicializa la lista de amigos, muestra los datos de la lista, presenta un menú de tres opciones ((1) Buscar por fecha; (2) Buscar por nombre; (3) Salir). Si el usuario elige la opción (1) pide una fecha y busca los amigos que cumplen años en esa fecha. Si el usuario elige la opción (2) pide un nombre y busca los amigos que tienen ese nombre. El programa le permite al usuario nuevas búsquedas hasta que selecciona salir.

```
#define DIM 20

int main(void)
{
    AMIGO amigos[DIM];
    FECHA f;
    char nombre[30];
    int opc;

    printf ("Seleccione una opcion:\n");
    printf ("(1) Buscar por fecha\n");
    printf ("(2) Buscar por nombre\n");
    printf ("(3) Salir\n");
    printf ("Opcion:\n");
    scanf("%d", opc);

    printf("Hasta luego\n");
}
}
```

PARTE III: (8 puntos)

1. (3 puntos) Para la gestión de los libros de una pequeña biblioteca es preciso conocer su título, su autor, el ISBN (ejemplo: 978-3-16-148410-0), el campo de estudio (biología, matemáticas, física, química, literatura) y si está prestado o no. Usando typedef y tipos enumerados donde sea posible, defina la estructura FICHALIBRO que permita almacenar los datos de cualquier libro. Declare una variable para manejar una colección de 1000 libros.

2. (5 puntos) Suponga que se tiene un archivo de texto llamado "prueba.txt" en el mismo directorio en que se encuentra el código fuente. Escriba un programa que abra el archivo en modo lectura, si hay un error en la apertura del archivo muestre un mensaje de error. Si no hay error escriba el contenido del archivo en un segundo archivo "nuevo.txt", eliminando todos espacios en blanco, hasta llegar al final del primer archivo.