

**Parcial 3 (25 pts)**

**PARTE I : Seleccione exactamente una alternativa correcta en las siguientes preguntas (4 pts):**

<p>1-¿Cuál es la salida del siguiente código?</p> <pre>char cad1[]="Venezuela"; char cad2[]; cad2=cad1; printf("%s", cad2);</pre>	<p>___ a) Venezuela.          ___ b) No se imprime nada.          ___ <b>c) Error de compilación.</b>          ___ d) Ninguna de las anteriores.</p>
<p>2-¿Cuál de estas declaraciones son correctas ?</p>	<p>___ a) enum = {red, blue=5};          ___ b) enum = {red; blue=5};          ___ <b>c) enum {red, blue=5};</b>          ___ d) La 3 opciones anteriores son correctas.          ___ e) Ninguna de las anteriores.</p>
<p>3) ¿Cuál es la salida del siguiente código?</p> <pre>int main() {     int a[3][4] = {1, 2, 3, 4, 4, 3, 2,     8, 7, 8,9, 0};     fun(a[1][2]);     return 0; } void fun(int i) {     printf("%d\n", i); }</pre>	<p>___ a) 1          ___ <b>b) 2</b>          ___ c) 3          ___ d) 4          ___ e) Ninguna de las anteriores.</p>
<p>4-¿Cuál es la salida del siguiente código?</p> <pre>char cad[]="venezuela"; int i; for (i=strlen(cad)-1; i&gt;0; i--) {     cad[i]=cad[strlen(cad)-1-i]; } printf("%s", cad);</pre>	<p>___ a) venezuela          ___ <b>b) venezenev</b>          ___ c) aleuzenev          ___ d) venezuel          ___ e) Ninguna de las anteriores.</p>

**PARTE II :**

Se requiere simular el sistema Capta Huellas para la venta de productos en los supermercados en Venezuela, el cual no permite la venta del mismo producto dos veces en la misma semana al mismo cliente. Los datos que se van a tomar en cuenta son :

- Cliente : **CI** (entero), **Nombres** (cadena), **Sexo** (Carácter que toma F o M) y **Huella** (cadena de ceros y unos con una longitud de 15).
- Producto : Código del producto (**CodProd**: entero), Nombre de producto (**NomProd**) y **Precio** (float).
- Venta : **CI** (entero), **Huella**(cadena de ceros y unos), Código del producto(**CodProd**), Fecha de

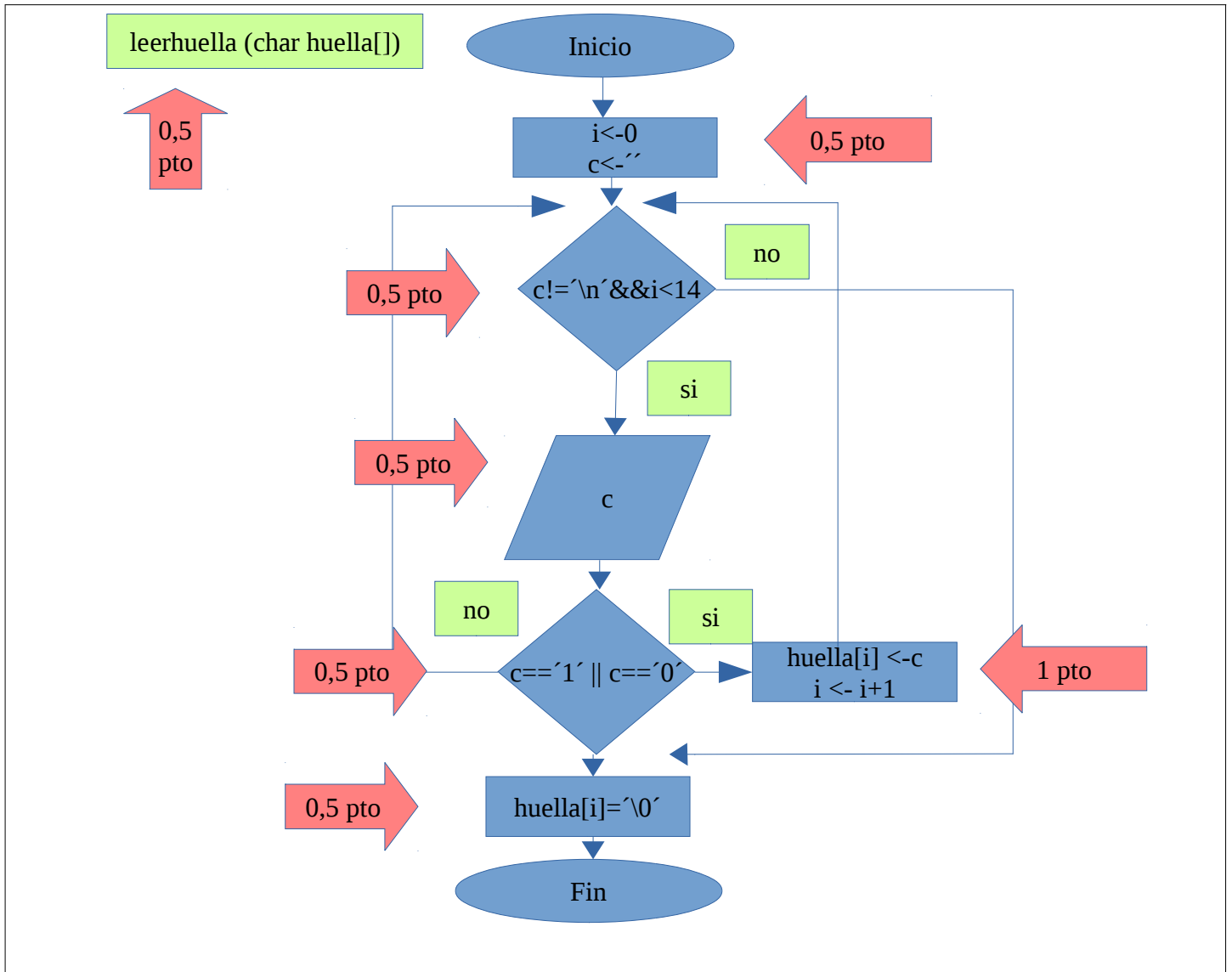
la venta (**FV**: estructura que contiene día, mes y año), **Unidad** (entero), Forma de pago (**FP**: cadena de caracteres). La forma de pago puede ser : Contado o Débito.

**a)** Realice la definición en C de las estructuras de datos necesarias para representar el sistema Capta Huellas. *NOTA: Se debe usar los mismos nombres de variables que están en negritas y subrayadas.*

**(5 pts)**

```
typedef struct {                                (1 pto)
    int CI;
    char nombre [20];
    char apellido[20];
    char sexo;
    char Huella[15];
}Cliente;
typedef struct {                                (1 pto)
    int CodPro;
    char NomProd[20];
    float precio;
}Producto;
typedef struct {                                (1 pto)
    int dia;
    int mes;
    int anyo;
}Fecha;
typedef struct {                                (2 pto)
    int CI;
    char Huella[15];
    int CodPro;
    Fecha fv;
    int unidad;
    char fp[7];
}Venta;
```

b) Realice un sub-programa para leer la *Huella* (Diagrama de Flujo), sabiendo que la *Huella* es una cadena de caracteres de longitud 15 y que toma solamente valores (1) o (0). **(4pts)**



d) Complete el siguiente procedimiento *Alerta* para verificar si se está vendiendo a la misma persona el mismo producto dos veces en la misma semana. Se supone que la lista de los productos vendidos para los clientes está ordenada en forma descendente por fecha. La variable **SI** debe devolver (1) si la diferencia entre la fecha de la venta actual y la fecha de la última venta es menos o igual a 7 días, o (0) en caso contrario.

Ejemplos para calcular la diferencia entre dos fechas (OJO: no tomen en cuenta el año) : **(7pts)**

<p><b>Ejemplo I:</b>            Fecha actual : 15 - 01 - 2016            Fecha última venta : 02 - 01 - 2016</p> <p>En este ejemplo el año y el mes son iguales en las dos fechas y el día de la la fecha actual es mayor que el de la fecha de la última venta, por lo tanto la diferencia es (15-02) 13 días y se le puede vender al cliente.</p>	<p><b>Ejemplo II:</b>            Fecha actual : 02 - 02 - 2016            Fecha última venta : 31 - 01 - 2016</p> <p>En este ejemplo el día de la fecha actual es menor que el día de la fecha de la última venta, por lo tanto se resta (1) al mes y se le suma 30 al día.            Días = (2+30) - 31 = 1            Mes = (2-1) = 1            La diferencia es 1 día, por lo tanto el cliente no puede realizar la compra.</p>
---	--

```

void Alerta (Venta V[], int n, int CI, char Huella[], int CodPro, Fecha F, int *SI)
// n es la longitud del arreglo de la venta.
// CI: Cédula de Identidad,
// V[] : es arreglo de estructura que contiene las ventas realizadas.
// huella : es la cadena de unos y ceros que representa la huella del cliente.
// CodPro : Es el nuevo código de product que se está vendiendo al cliente.
// F : Es la fecha actual de la venta.
// SI : Es la variable que va a tener (1) si se está vendiendo el mismo producto al mismo cliente en menos o
igual a 7 días.
{
    int i;
    int comparar;
    int dif_dias;
    For (i=0; i<n; i++) (0,5 pto)
    {
        dif_dias=30;
        *SI=0;
        comparar=strcmp(V[i].huella, huella);
        if (V[i].fv.dia==F.dia && V[i].fv.mes==F.mes) (1 pto)
            {
                dif_dias=0;
            }
        else if (V[i].fv.dia<F.dia && V[i].fv.mes==F.mes) (1 pto)
            {
                dif_dias= F.dia- V[i].fv.dia;
            }
        else if (V[i].fv.dia>F.dia && (V[i].fv.mes+1)==F.mes) (1 pto)
            {
                dif_dias= (F.dia+30)- V[i].fv.dia;
            }
        else if ((F.mes-V[i].fv.mes)>1 ) (1 pto)
            {
                dif_dias=30;
            }

        If (V[i].CI==CI && comparar==0) (1 pto)
        {
            If (V[i].CodPro==CodPro && dif_dias<=7) *SI=1; (1 pto)
            Break; (0,5 pto)
            // para salir del ciclo porque la lista de venta está
            ordenada en forma descendente, por lo tanto no hace falta
            seguir hasta el final.
        }
    }
}

```

**d)** Realice un sub-programa en C que reciba la *Huella* como vector de caracteres con tamaño 15 y elimine de dicho vector, los elementos repetidos y reajuste el arreglo colocando al final del reajuste '1' imprimir ambos vectores (El vector original de la Huella y el nuevo vector). **(5pts)**

Ejemplo:

Entrada (Huella): 10101111001010

Salida : 10101010101 (El último **uno** es agregado después del reajuste del vector de Huella).

```
void reajustar (char huella[])           (0,5 pto)
{
    int i, n, j=0;
    n=strlen(huella);                    (0,5 pto)
    char nuevo[n+1];                      (0,5 pto)
    for (i=0; i<n ; i++)                  (0,5 pto)
        if (huella[i]!=huella[i+1])      (0,5 pto)
            {
                nuevo[j]=huella[i];        (0,5 pto)
                j=j+1;                      (0,5 pto)
            }
    nuevo[j]='1';                          (0,5 pto)
    nuevo[j+1]='\0';                        (0,5 pto)
    puts("\n");
    puts(huella);                           (0,25 pto)
    puts("\n");
    puts(nuevo);                             (0,25 pto)
}
```