

TRABAJO PRÁCTICO: Agencia de Turismo Toursven

ASIGNATURA: COMPUTACIÓN I

CÓDIGO: 323

FECHA DE ENTREGA AL ESTUDIANTE: Adjunto a la Primera Prueba Parcial

FECHA DE DEVOLUCIÓN: Adjunto a la Prueba Integral

NOMBRE DEL ESTUDIANTE: DANIEL JESUS RONDON SALAZAR

CÉDULA DE IDENTIDAD: 22854574

CENTRO LOCAL: ANZOATEGUI

CARRERA: 126

NÚMERO DE ORIGINALES:

FIRMA DEL ESTUDIANTE:

LAPSO: 2015/2

RESULTADOS

OBJ #	5	6	7	8
0: NL 1:L				

La Solución analítica del Problema.

En la solución de este problema se debe conocer las características del lenguaje a utilizar, entre ellas sus limitaciones, y de las técnicas o recursos de programación, arreglos, registros, ciclos, entre otros; y además de eso buscar rutinas o procedimientos hechos en otros programas que puedan adaptarse a la solución de éste.

Datos de Entrada.

- 1.Nombre y Apellido
- 2.Numero de Pasaporte.
- 3.Origen de Pasaporte (V/E).
- 4.Fecha de expedición del pasaporte.
- 5.Fecha de expedición del certificado de vacuna.
- 6.Fecha de expedición de la declaración de impuestos.

Datos de Salida.

Incluye los datos de entrada, un listado ordenado de los clientes e información relacionada a la cantidad de clientes en el sistema.

Proceso:

- Se introducen los datos.
- Se muestran los datos introducidos.
- Se guardan los datos.
- Se consultan los datos de salida.
- Se pueden modificar los datos o eliminarlos.

UNIDAD 5. La solución del problema utilizando la metodología MAPS.

ETAPA I: El Diálogo.

Una agencia de turismo llamada TOURSVEN necesita de un sistema automatizado que le permita tener un registro de todos sus clientes.

¿Qué datos se exigen de cada cliente?

Cada cliente debe dar estos datos: nombre y apellido, número de pasaporte, origen de pasaporte, fecha e expedición del pasaporte, fecha del certificado de vacuna y la fecha de la declaración de impuestos.

¿Cómo estará estructurado principalmente el menú de opciones?

Se estructurará en 5 opciones principales: Incluir, Modificar, Eliminar, Reporte y Salida.

¿Qué otras informaciones se requieren?

Aparte de los datos de los clientes se necesita saber si este tiene los recaudos vigentes, y otras informaciones incluidas en la submenú Reporte, como:

- cantidad de Turistas Venezolanos
- cantidad de Turistas extranjeros

- cantidad de Turistas con c. de vacuna vencido
- cantidad de Turistas con d. de impuestos vencido
- cantidad de Turistas con recaudos vigentes.

ETAPA II: ESPECIFICACIONES

Datos de Entrada:

- 1.Nombre y Apellido: El nombre y apellido solo debe contener caracteres alfabéticos y espacios.
- 2.Numero de pasaporte: Debe ser un número entero positivo.
- 3.Origen de pasaporte: Sólo admite la opción[V]venezolano o [E]extranjero
- 4.Fecha de pasaporte,
- 5.Fecha de vacuna,
- 6.Fecha de declaracion de impuestos: las fechas deben ser introducida en formato d/m/aa.

Datos de Salida

- 1.Datos de entrada
 - 1.2.Mensaje: "Recaudos no vigentes" , aparecerá cuando el cliente tenga uno de sus requisitos vencidos.
 {((Origen de pasaporte = V "y" Fecha de expedicion de pasaporte > un año) "o" (origen de pasaporte =E y fecha de expedicion de pasaporte > dos años))
 "o" fecha de vacuna > un año
 "o" fecha declaracion de impuestos > un año}
- 2.Listado Ordenado de los clientes.
 {Orden por numero de pasaporte seguido del nombre y apellido
 Numero de pasaporte Nombre y apellido}
- 3.Otras informaciones:
 - cantidad de Turistas Venezolanos. {origen de pasaporte = V}
 - cantidad de Turistas extranjeros. {Origen de pasaporte = E}
 - cantidad de Turistas con c. de vacuna vencido {fecha de vacuna > un año}
 - cantidad de Turistas con d. de impuestos vencido {fecha declaracion de impuestos > un año}
 - cantidad de Turistas con recaudos vigentes.
 {((Origen de pasaporte = V "y" Fecha de expedicion de pasaporte < un año)
 "o" (origen de pasaporte =E y fecha de expedicion de pasaporte < dos años))
 "y" fecha de vacuna < un año
 "y" fecha declaracion de impuestos < un año}

ETAPA III: LA PARTICION

El proposito del algoritmo es hacer un menú de 5 opciones principales:

- 1.Incluir
- 2.Modificar
- 3.Eliminar
- 4.Reporte
- 5.Salida

Cada opción a su vez tendrá su propia subdivisión:

- 1.Incluir:
 - 1.1.Ingrese Nombre y Apellido
 - 1.2.Ingrese numero de pasaporte
 - 1.3.Ingrese origen de pasaporte
 - 1.4.Ingrese fecha del pasaporte
 - 1.5.Ingrese fecha del certificado de vacuna
 - 1.6.Ingrese fecha de la declaracion de impuestos
- 2.Modificar.
 - 2.1.Ingrese numero de pasaporte.
 - 2.2.Menu de modificacion
 - 2.2.1.Nombre y Apellido.
 - 2.2.2.Origen de pasaporte.
 - 2.2.3.Fecha del pasaporte
 - 2.2.4.Fecha certificado de vacuna
 - 2.2.5.Fecha declaracion de impuestos
- 3.Eliminar
 - 3.1.Ingrese numero de pasaporte
 - 3.2.Datos
 - 3.2.1.Nombre y Apellido
 - 3.2.2.Numero de pasaporte
 - 3.2.3.Origen de pasaporte
 - 3.2.4.Fecha del pasaporte
 - 3.2.5.Fecha certificado de vacuna
 - 3.2.6.Fecha declaracion de impuestos
 - 3.2.Esta seguro que desea eliminar este registro?[S/N]
- 4.Reporte
 - 4.1.Listado Ordenado
 - 4.2.Consultar
 - 4.2.1.Ingrese numero de pasaporte
 - 4.2.2.Datos
 - 4.2.2.1.Nombre y Apellido
 - 4.2.2.2.Numero de pasaporte
 - 4.2.2.3.Origen de pasaporte
 - 4.2.2.4.Fecha del pasaporte
 - 4.2.2.5.Fecha certificado de vacuna
 - 4.2.2.6.Fecha declaracion de impuestos
 - (4.2.2.7.Mensaje:"Recaudos no vigentes")
 - 4.3.Informacion-Contadores
 - 4.3.1.Turistas Venezolanos

- 4.3.2.Turistas extranjeros
- 4.3.3.Turistas con c. de vacuna vencido
- 4.3.4.Turistas con d. de impuestos vencido
- 4.3.5.Turistas con recaudos vigentes.
- 4.4.Volver al menu.

5.Salida

ETAPA IV: DEFINICION DE ABSTRACCIONES.

Para la elaboración de este algoritmo se usará Freepascal. Se usarán principalmente la unidad predeterminada system, además de las unidades sysutils, dateutils y crt. Aparte de eso se empleará a lo largo de programa la funcion "existe archivo":

1.De la unidad system se utilizará principalmente:

- Lenght: da un valor numerico de la cantidad de caracteres en una cadena.
- Filesize: Cuenta la cantidad de registros en F.
- Val: convierte un string a un valor numerico (byte, integer, longint,real),
- Uppcase: convierte una cadena completa a mayúsculas.
- Ioresult: contiene el resultado de cualquier llamada de entrada / salida, cuando la directiva del compilador {\$I-} está activa, desactiva la comprobación del IO. Si el resultado es cero, la operación se ha completado con éxito. Si no es cero, se produjo un error.
- Assign: Le da un nombre a la variable f, que puede ser cualquier tipo de archivo. Este comando no abre el archivo, solo le da un nombre, y marca el archivo como cerrado.
- seek: Permite posicionarse en el numero de un registro. Se debe tener en cuenta que el primer registro tiene contador= 0, y el ultimo = filesize(f).
- write: Escribe el contenido de las variables en el fichero F.
- rewrite: crea un archivo F, si el archivo existe le da un tamaño de longitud 0,
- rename: Permite cambiar el nombre de un archivo ya asignado.
- Close: Cierra el archivo
- Reset: abre un archivo para su lectura.
- Erase: remueve un archivo cerrado del disco duro.

2. De la unidad sysutils se utilizará:

- Datetostr: Pasa una variable TDate a su valor en string.
- Date: Determina la fecha actual, basandose en la fecha del sistema operativo.
- Trystrtodate: TryStrToDate permite convertir un string a un valor de fecha TDateTime, y almacena la fecha en Valor. La fecha debe constar de 1 a tres dígitos, separados por el carácter separador de fecha. Si se dan dos números, se supone que deben formar el día y el mes del año en curso. Si sólo se da un número, que se supone que representa el día del mes en curso.

El orden de los dígitos (a / m / d, m / d / a, d / m / a) se determina a partir de la variable ShortDateFormat.

La función devuelve True si la cadena contiene una indicación de fecha válida, False en caso contrario.

3. De la unidad dateutils se utilizará:

- Daysbetween: Determina el numero de dias que hay entre dos fechas TDate.
- Incyear: Permite agregarle un año e incluso restarle un año a una fecha TDate.

4. De la unidad crt se utilizará:

- Clrscr: Clearscreen borra la ventana actual, y establece el cursor en la esquina superior izquierda de la ventana actual.
- Readkey: Permite leer un caracter enviado desde el teclado.

5. function existearchivo : boolean;

```
begin
  assign(f,archivo);
  {$I-} reset(f); {$I+}
  if iorresult <> 0 then
    existearchivo := false
  else
    existearchivo := true;
end;
```

ETAPA V: CODIFICACION.

En esta etapa se elabora el desarrollo del algoritmo en el lenguaje Pascal, en este caso en el

compilador Free Pascal. Ya que este punto es correspondiente con el Objetivo 8 no se desarrollará en esta unidad, para así evitar repeticiones innecesarias.

ETAPA VI: PRUEBA Y VERIFICACION.

En esta etapa se prueba el funcionamiento del algoritmo en Free Pascal, ejecutándolo ingresando datos y verificando las salidas, determinando si estas satisfacen las postcondiciones pautadas.

ETAPA VII: PRESENTACION

En esta etapa se añaden los comentarios necesarios al sistema para identificar la fecha, nombre del autor, e identificar el propósito de las rutinas empleadas en el programa.

Unidad 6 PSEUDOCODIGO EN ESPAÑOL

Algoritmo turismotoursven

tipos

```
TElturista = registro
Numeropas : entero_largo //numero de pasaporte
Nombreape : cadena      //nombre y apellido
origenpas : caracter     //origen de pasaporte
fechapas,      //Fecha Exp. pasaporte
fechavac,      //Fecha Exp. certificado vacuna
fechadec: Tipo fecha    //Fecha Exp. declaracion de impuestos
fin_TElturista
```

variables

```
Archivo      : fichero de TElturista
Archivo_Auxiliar : fichero de TElturista
datos: TElturista //auxiliar para guardar los datos
i,j,k :entero    //variables indice
```

funcion existearchivo : booleano

```
inicio_funcion
  Intentar_abrir Archivo
  si Archivo fue abierto entonces
    existearchivo := cierto
  sino
    existearchivo := falso
fin_funcion
```

funcion Expedicion (fecha:Tipo fecha) : entero

```
Inicio_funcion
  Diasentre (Fechaactual,Fecha)
```

fin_funcion

```
function anhos(num:entero) : entero
begin
  Diasentre(FechaActual ,Decrementar_año(FechaActual,num));
end;
```

procedimiento guardardatos (variable aux:TElturista)

```
Inicio_procedimiento
  si existearchivo = cierto entonces
    inicio_si
      buscar longitud_de_archivo
    fin_si
  sino
    inicio_sino
      crear Archivo
    end;
    grabar aux
    cerrar Archivo
    Escribir ' Datos guardado Pulse una tecla'
    leer_caracter
  fin_procedimiento
```

procedimiento pasaportevalido(variable num:entero_largo)

```
variables
  strnum :cadena;    //auxiliar del numero de pasaporte
  valido : booleano; //Booleano para validar el numero de pasaporte
  aux: TElturista;
```

inicio_procedimiento

```
  repetir
    repetir
      leer strnum
      valor (strnu,num)
      si num < 1 entonces
        inicio_si
          Escribir '      *Por favor,introduzca solo numeros*      : '
        fin_si
      hasta_que num >= 1
      valido := cierto
      si existearchivo = cierto entonces
        inicio_si
          desde i := 0 hasta (longitud_de_Archivo) - 1 hacer
            inicio_desde
              buscar i
              leer aux
```



```

        Si aux.numeropas = num entonces
            inicio_si
                valido := cierto
                Escribir '      Este numero ya esta en el sistema,      '
                Escribir '      *vuelva a introducirlo*      : '
            fin_si
        fin_desde
    cerrar Archivo
fin_si
hasta_que valido=cierto
fin_procedimiento

```

```

procedimiento origenvalido (variable opcion:caracter)
    inicio_procedimiento
        repetir
            opcion:= Mayuscula(leer_caracter)
        hasta_que opcion pertenece ['E','V']
        Escribir opcion
    fin_procedimiento

```

```

procedimiento fechavalida ( variable fecha:Tipo fecha)
    variable
        valido:booleano
        strfecha:cadena
    inicio_procedimiento
        valido:=falso
        repetir
            Leer strfecha
            valido:= Cadena_a_Tipofecha (strfecha,fecha)
            Si valido = false entonces
                Inicio_si
                    Escribir '      *por favor,fecha en formato DD/MM/AAAA*      '
                    Escribir '      Alternativa 1: DD/MesActual/AñoActual      '
                    Escribir '      Alternativa 2: DD/MM/AñoActual      '
                    Escribir '      Alternativa 3: DD/MM/20AA      '
                    Escribir '      Introduzca nuevamente la fecha      : '
                Inicio_si
            hasta_que valido=cierto
        fin_procedimiento

```

```

procedimiento mostrardatos;
    inicio_procedimiento
        Escribir ' Nombre y Apellido      = ',Nombreape
        Escribir ' Numero de pasaporte      = ',numeropas
        Escribir ' Origen de pasaporte      = ',origenpas
        Escribir ' Fecha del pasaporte      = ',fechapas
        Escribir ' Fecha Certificado de vacuna      = ',fechavac
        Escribir ' Fecha Declaracion de impuestos = ',fechadec
    fin_procedimiento

```

```

procedure NombreYapellidovalido (variable nomb:cadena)
tipo
  TAscii = Conjunto_ASCII //contiene 255 caracteres
var
  alfabeto,espacio:TAscii;
  valido : boolean;    //Booleano para validar el nombre y apellido

begin
  alfabeto:=[#65..#90,#97..#122,#165..#166]
  espacio :=[#32]
  repeat
    valido := falso
    leer nomb
    desde k := 1 hasta longitud(nomb) hacer
      inicio_desde
        si nomb[k] pertenece alfabeto U espacio entonces
          valido := cierto
      fin_desde
    Si valido = false entonces
      inicio_si
        Escribir '    *Por favor,introduzca solo letras y espacio* : '
      fin_si
    hasta_que (valido = true) y (k = longitud(nomb))
fin_procedimiento

```

```

procedimiento entradadatos;
var
  opcion : caracter;    //caracter de opciones

begin

  Escribir ' Entrada Datos turista'
  Escribir;
  Escribir ' 1) Ingrese  Nombre  y  Apellido          : '
  //En este campo solo se admiten letras y/o espacios
  Rutina Nombreyapellidovalido(nombreape)

  Escribir ' 2) Ingrese  Numero  pasaporte            : '
  //En este campo solo se admiten numeros
  //Este campo no admitira 2 numeros de pasaporte iguales
  Rutina pasaportevalido( Numeropas)

  Escribir ' 3) Ingrese  Origen  del  pasaporte  [V/E] : '
  //La persona es venezolana o es extranjera
  //V= Venezolano  E=Extranjero}
  Rutina origenvalido(origenpas)

```

Escribir ' 4) Ingrese fecha del Pasaporte : '
//La fecha debe ser en formato D/M/AA}
Rutina fechavalida(fechapas)

Escribir ' 5) Ingrese fecha del certificado de vacuna : '
//La fecha debe ser en formato D/M/AA
Rutina fechavalida(fechavac)

Escribir ' 6) Ingrese fecha de la declaracion de impuestos : '
//La fecha debe ser en formato D/M/AA
Rutina fechavalida(fechadec)

Escribir ' Ha introducido los siguientes datos: '

Escribir;

Mostrar datos

Escribir;

Escribir;

Escribir ' Desea Guardar Los Datos? '

Escribir;

Escribir ' Pulse [S/N]'

repetir

opcion := Mayuscula(leer_caracter)

hasta_que opcion pertenece ['S','N']

Si opcion = 'S' entonces

guardar datos

fin_procedimiento

procedimiento busqueda

variables

num : entero_largo; //auxiliar del numero de pasaporte

encontrado : booleano; //determina la existencia de un registro

Inicio_procedimiento

Si existearchivo = false then

Inicio_si

Escribir ' Error De Archivo O No Existe Pulse Una Tecla'

Leer_tecla

fin_si

Sino

Inicio_sino

Escribir ' Buscar Un turista '

Escribir

Escribir ' Entre Numero de pasaporte : '

Leer num

encontrado := false

Desde i := 0 hasta (longitud_de_archivo) - 1 hacer

Inicio_desde

Buscar i

Leer datos

```

    Si numeropas = num entonces
        inicio_si
            encontrado := true
        fin_si
    fin_desde
    Si encontrado = true entonces
        inicio_si
            Escribir
            mostrar datos
            Escribir;
            Si ((Expedicion(fechapas) > anhos(1) y Origenpas='V')
                o (Expedicion(fechapas) > anhos(2) y Origenpas='E'))
                o Expedicion(fechavac) > anhos(1)
                o Expedicion(fechadec) > anhos(1) entonces
                    Escribir ' "RECAUDOS NO VIGENTES"'
                    Escribir;
                    Escribir ' Pulse una tecla '
                    Escribir;
                fin_si
            Sino
                Inicio_sino
                    Escribir ' No se encuentran datos del turista '
                    Escribir
                    Escribir ' Pulse Una Tecla '
                    Escribir
                fin_sino;
            Cerrar archivo;
        fin_sino
    fin_procedimiento

```

```

procedure ordenar;
// Existe un famoso metodo llamado Burbuja
// Con este metodo se ordenaran los datos por el numero de pasaporte
variables
    Copia, Aux : TElturista; //copias de TElturista
Inicio
    Si existearchivo = falso entonces
        Inicio_si
            Escribir ' Error De Archivo O No Existe Pulse Una Tecla'
            Leer_caracter;
        fin_si
    Sino
        Inicio_sino
            desde i := 0 hasta (longitud_de_archivo) - 1 hacer
                Inicio_desde
                    buscar i
                    leer datos
                    desde j := i + 1 hasta (longitud_de_archivo) - 1 hacer
                        inicio_desde

```

```

        buscar j
        leer copia
        Si datos.Numeropas > Copia.Numeropas entonces
            inicio_si
                Aux := datos
                datos := Copia
                Copia := Aux
                buscar i
                grabar datos
                buscar j
                grabar copia
            fin_si
        fin_desde;
    fin_desde;
fin_sino;
cerrar archivo
fin_procedimiento

```

```

procedure modificacion;
var
    num : entero_largo //auxiliar del numero de pasaporte
    encontrado : boolean; //determina la existencia de un registro
    opcion : caracter; //caracter de opciones

```

```

Inicio_procedimiento
Si existearchivo = falso entonces
    inicio_si
        Escribir ' Error De Archivo O No Existe Pulse Una Tecla'
        Leer_caracter;
    fin_si
sino
    inicio_sino
        Escribir ' Entre numero de pasaporte : '
        Leer num
        encontrado := falso
        Desde i := 0 hasta (longitud_de_archivo) - 1 hacer
            Inicio_desde
                buscar i
                leer datos
                Si Numeropas = num entonces
                    inico_si
                        encontrado := cierto
                    fin_si;
            fin_desde;
        Si encontrado = cierto entonces
            begin
                repetir
                    Escribir ' **** Modificacion De Datos ****'

```

```

Escribir
Escribir ' Numero de pasaporte ' ,numeropas
Escribir;
Escribir ' [1] = Nombre y Apellido ' ,Nombreaape
Escribir ' [2] = Origen del pasaporte ' ,origenpas
Escribir ' [3] = Fecha del pasaporte ' ,fechapas
Escribir ' [4] = Fecha Certificado vacuna ' ,fechavac
Escribir ' [5] = Fecha declaracion de impuestos ' ,fechadec
Escribir ' [6] = finalizar');
Escribir;
repetir
    opcion := leer_caracter;
    hasta_que opcion pertenece ['1','2','3','4','5','6']
segun_sea opcion hacer
'1' : Inicio_opcion 1
    Escribir ' Ingrese Nombre y Apellido ' : '
    Rutina Nombreyapellidovalido(nombreaape)
    buscar i
    grabar datos
    fin_opcion 1

'2' : Inicio_opcion 2
    Escribir ' Ingrese Origen del pasaporte [V/E] ' : '
    Rutina origenvalido(origenpas)
    buscar i
    grabar datos
    fin_opcion 2

'3' : Inicio_opcion 3
    Escribir ' Ingrese fecha del Pasaporte ' : '
    Rutina fechavalida(fechapas)
    buscar i
    grabar datos
    fin_opcion 3

'4' : Inicio_opcion 4
    Escribir(' Ingrese fecha del certificado de vacuna ' : '
    Rutina fechavalida(fechavac)
    buscar i
    grabar datos
    Fin_opcion 4

'5' : Inicio_opcion 5
    Escribir ' Ingrese fecha de la declaracion de impuestos ' : '
    Rutina fechavalida(fechadec)
    buscar i
    grabar datos
    Inicio_opcion 5
fin_segun
hasta_que opcion = '6'
fin_si
sino

```

```

inicio_sino
    Escribir ' Dato No Encontrado Pulse Una Tecla'
    leer_caracter
fin_sino
cerrar Archivo
fin_sino
fin_procedimiento;

```

procedimiento anularegistro

```

variables
    num      : entero_largo      //auxiliar del numero de pasaporte
    encontrado: booleano         //determina si un registro existe
    opcion : caracter;           //caracter de opciones

```

Inicio

Si existearchivo = falso entonces

```

    Inicio_si
        Escribir ' Error de Archivo o no existe Pulse Una Tecla'
        Leer_tecla
    fin_si

```

sino

```

    inicio_sino
        Escribir ' Entre numero de pasaporte : '
        Leer num
        encontrado := falso
        Desde i := 0 hasta (longitud_de_archivo) - 1 hacer

```

```

            inicio_desde
                buscar i
                leer datos
                Si Numeropas = num entonces
                    Inicio_si
                        Escribir
                        mostrar datos_guardados
                        Escribir
                        encontrado := true
                        j := i;
                    fin_si
            fin_desde

```

Si encontrado = cierto entonces

```

    Inicio_si
        Escribir ' <<< Esta Seguro Que Desea Eliminar Este registro? [S/N] >>>'
        repetir

```

```

            opcion := Mayuscula(leer_tecla);
            hasta_que opcion pertenece ['S','N'];
            Si opcion = 'S' entonces

```

```

                Inicio_si
                    Crear Archivo_Auxiliar
                    Desde i := 0 hasta (longitud_de_Archivo) - 1 hacer
                        Inicio_desde

```

```

        buscar i
        Si i <> j entonces
            Inicio_si
                leer datos
                grabar datos en Archivo_Auxiliar
            fin_si
        fin_desde
        Cerrar Archivo
        cerrar Archivo_Auxiliar
        borrar Archivo
        Cambiar_Nombre Archivo_Auxiliar por Archivo
        Escribir ' Datos eliminado Pulse una tecla'
        leer_caracter
        end;
        Si opcion = 'N' entonces
            inicio
                cerrar Archivo
            fin
        fin_si
    sino
        inicio_sino
            Escribir ' Datos no encontrado Pulse una tecla'
            leer_caracter
            Cerrar Archivo
        fin_sino
    fin_sino
fin_si

```

```

procedimiento mostrar
    Inicio_procedimiento
        Si existearchivo = falso entonces
            Inicio_si
                salir
            Fin_si
        sino
            inicio_sino
                Escribir ' <<< Listado Ordenado >>> '
                Escribir
                Desde i := 0 hasta (longitud_de_Archivo) - 1 hacer
                    inicio_desde
                        buscar i
                        leer datos
                        Si Numeropas <> 0 entonces
                            Inicio_si
                                Escribir ' ',datos.Numeropas,', ',Nombreape
                            fin_si
                        fin_desde
                    cerrar Archivo
                Escribir
            fin_sino
        fin_sino
    fin_procedimiento

```



```
    Escribir ' <<< Pulse Una Tecla Para Regresar >>>'
    Leer_caracter
    fin_desde
fin_procedimiento
```

procedimiento contador

variables

```
    vene,      //Contador de turistas venezolanos
    extra,     //contador de turistas extranjeros
    vacun,     //contador de turistas con fecha de vacuna vencida
    impue,     //contador de turistas con fecha de declaracion vencida
    recau:integer; //contador de turistas con recaudos vigentes
```

Inicio_procedimiento

Si existearchivo = falso entonces

inicio_si

Escribir ' Error De Archivo O No Existe Pulse Una Tecla '

leer_caracter

fin_si

sino

inicio_sino

vene := 0

extra := 0

vacun := 0

impue := 0

recau := 0

desde i := 0 hasta (longitud_de_Archivo) - 1 hacer

Inicio_desde

buscar i

leer datos

Si Numeropas <> 0 entonces

inicio_si

Si origenpas = 'V' entonces

vene := vene + 1

Si origenpas = 'E' entonces

extra := extra + 1

si Expedicion(fechavac) > anhos(1) entonces

vacun := vacun + 1

si Expedicion(fechadec) > anhos(1) entonces

impue := impue + 1

si ((Expedicion(fechapas) <=anhos(1) y Origenpas='V')

o (Expedicion(fechapas)<=anhos(2) y Origenpas='E'))

y Expedicion(fechavac)<=anhos(1)

y Expedicion(fechadec)<=anhos(1) entonces

recau := recau + 1;

fin_si;

fin_desde;

Escribir ' <<< Datos>>>'

Escribir

Escribir ' turistas venezolanos ', vene

```

    Escribir ' turistas extranjeros      ',extra
    Escribir ' turistas con c. de vacuna vencido ',vacun
    Escribir ' turistas con d. de impuestos vencido ',impue
    Escribir ' turistas con Recaudos vigentes    ',recau
    Cerrar Archivo
    Escribir
    Escribir ' <<< Pulse Una Tecla Para Regresar >>>'
    Leer_caracter
fin_sino
fin_procedimiento;

```

procedimiento menu;

```

variables
    sal : booleano //booleano para salir del sistema
    opcion : caracter //caracter de opciones
inicio_procedimiento
    sal := falso
    repetir
        Escribir ' ***** Menu General *****'
        Escribir
        Escribir ' 1 = INCLUIR'
        Escribir ' 2 = MODIFICAR'
        Escribir ' 3 = ELIMINAR'
        Escribir ' 4 = REPORTE'
        Escribir ' 5 = SALIDA'
        Escribir
        Escribir ' >>> Elija Opcion <<<'
        repetir
            opcion := leer_caracter
        hasta_que opcion pertenece ['1','2','3','4','5']
        Segun_sea opcion hacer
            '1' : entradadatos
            '2' : modificacion;
            '3' : anularegistro;
            '4' : Inicio_opcion 4
        repetir
            Escribir ' Pulse L = LISTADO ORDENADO'
            Escribir ' Pulse C = CONSULTA'
            Escribir ' Pulse I = INFORMACION-CONTADORES'
            Escribir ' Pulse V = VOLVER AL MENU '
            Escribir
            Escribir '<< Elija una opcion >>'
        repetir
            opcion := Mayuscula(leer_caracter);
        hasta_que opcion pertenece ['L','C','I','V'];
        segun_sea opcion hacer
            'L': Inicio_opcion L
                ordenar
                mostrar

```

```

        Fin_opcion L
        'C': busqueda
        'I': Contador
        fin_según
        hasta_que opcion = 'V'
        Fin_opcion 4

```

```

        '5' : sal := cierto;
        fin_según
        hasta_que sal = cierto;
        fin_procedimiento

```

```

Inicio
    menu
Fin_algoritmo.

```

UNIDAD 7 ROBUSTEZ DEL ALGORITMO

En la elaboración de un programa siempre se deben hacer correcciones para que este pueda lograr los resultados de salida deseado. Habiendo creado un programa eficaz después de muchas correcciones, también este debe ser amigable, es decir indicar al usuario por medio de un mensaje que no ha introducido los datos correctamente y darle una oportunidad de volver a introducir el dato en el campo correspondiente. Para cumplir con el concepto de robustez y amigabilidad recurrí a 5 procedimientos para esto:

1) procedure mensaje: La intención de esta rutina es dar un mensaje al usuario, antes de ingresar los datos, de las condiciones que se exigen en cada campo. También explica las alternativas que permite el programa para introducir la fecha en formato DD/MA/AAAA. Algoritmo en lenguaje pascal:

```

procedure mensaje;
begin
    writeln(' Nota: Antes de continuar es importante que lea esto: ');
    writeln;
    writeln(' -El campo 1:Nombre y apellido, admite solo letras y espacios. ');
    writeln(' -El campo 2:Numero de pasaporte, admite solo numeros. ');
    writeln(' -El campo 3:Origen de pasaporte, admite solo una letra V/E. ');
    writeln(' -El formato de fecha en los campos 4,5 y 6 es: DD/MM/AAAA . ');
    if Monthof(date) in [1..9] then
        writeln('      Alternativa 1: DD/', '0', MonthOf(date), '/', YearOf(date)  );
    else
        writeln('      Alternativa 1: DD/', '', MonthOf(date), '/', YearOf(date)  );
        writeln('      Alternativa 2: DD/MM/', YearOf(date), '  ');
        writeln('      Alternativa 3: DD/MM/20AA      ' );
        writeln;
        writeln('      Pulse [S] continuar [N] Volver al menu  ');
    end;
end;

```

2) Procedure NombreYapellidovalido: Esta rutina se encarga de validar el campo nombre y apellido. Su función principal es evitar la introducción de caracteres numéricos, aceptando solo las letras del alfabeto y el caracter espacio. Además para cumplir con el concepto de amigabilidad contiene un mensaje en caso de introducir un dato que no se ajustan a las precondiciones de entrada. A continuación el algoritmo en lenguaje pascal:

```
procedure NombreYapellidovalido(var nomb:string);

type
  TAscii = set of #1..#255;
var
  alfabeto,espacio:TAscii;
  valido : boolean;      {Booleano para validar el nombre y apellido}

begin
  alfabeto:=[#65..#90,#97..#122,#165..#166];
  espacio :=[#32];
  repeat
    valido := false;
    readln(nomb);
    for k := 1 to length(nomb) do
      begin
        if nomb[k] in alfabeto+espacio then
          valido := true;
        end;
        if valido = false then
          begin
            write('    *Por favor,introduzca solo letras y espacio* : ');
          end;
        until (valido = true) and (k = Length(nomb));
      end;
    end;
```

3) Procedure pasaportevalido: Esta rutina se encarga de validar entradas que solo contengan caracteres numéricos, también evita que el usuario entre un número de pasaporte ya registrado en el sistema, además esta rutina cumple con el concepto de amigabilidad. Algoritmo en lenguaje pascal:

```
procedure pasaportevalido(var num:longint);
var
  strnum:string;      {auxiliar del numero de pasaporte}
  valido : boolean;   {Booleano para validar el numero de pasaporte}
  aux: TElturista;

begin
  repeat
    repeat
      readln(strnum);
```

```

val(strnum,num);
if num < 1 then
begin
write('      *Por favor,introduzca solo numeros*      : ');
end;
until num >= 1;
valido := true;
if existearchivo=true then
begin
for i := 0 to filesize(f) - 1 do
begin
seek(f,i);
read(f,aux);
if aux.numeropas = num then
begin
valido := false;
writeln('      Este numero ya esta en el sistema,      ');
write('      *vuelva a introducirlo*      : ');
break;
end;
end;
close(f);
end;
until (valido = true);
end;

```

4) Procedure Origenvalido: Esta rutina valida el campo origen de pasaporte. Como condiciones de entrada se estableció que sólo se debe admitir un sólo caracter, tomando en cuenta que la persona puede ser de origen venezolano o extranjero. Si es venezolano el usuario debe introducir V, si es extranjero, E. Para cumplir con el concepto de amigabilidad en el mismo campo se da las condición de entrada: ' 3) Ingrese Origen del pasaporte [V/E] : '. Algoritmo en lenguaje pascal:

```

procedure origenvalido(var tec:char);
begin
repeat
tec:= upcase(readkey);
until tec in['E','V'];
writeln(tec);

```

5) Procedure fechavalida: Esta rutina acepta fechas introducidas que sean validas en formato D/M/AA. Un algoritmo para validar una fecha se complica cuando se ve que no todos los meses tienen 30 días, y cada cuatro años existe uno que es bisiesto. En pascal existe la función "trystrtodate" de la librería "sysutils", una función muy eficaz, con esto se evita crear una rutina compleja para este caso. Hay que recordar que la función trystrtodate permite alternativas en la introducción de una fecha:

Alternativa 1: DD/Mes actual/Año actual. Sólo con introducir el día el programa supone que es un día del mes actual.

Alternativa 2: DD/MM/Año actual. Introduciendo el día y el mes el programa supone que son del año actual

Alternativa 3: DD/MM/20AA. Sólo basta con introducir los últimos 2 dígitos del año actual.

Algoritmo en lenguaje pascal robusto y amigable.

```
procedure fechavalida( var fecha:TDate);
var
  valido:boolean;
  strfecha:string;
begin
  valido:=false;
  repeat
    ReadLn(strfecha);
    valido:=TryStrToDate(strfecha,fecha);
    if valido = false then
      begin
        writeln('      *por favor,fecha en formato DD/MM/AAAA*   ');
        if Monthof(date) in [1..9] then
          writeln('      Alternativa 1: DD/','0', MonthOf(date),'/',YearOf(date)   )
        else
          writeln('      Alternativa 1: DD/',' ', MonthOf(date),'/',YearOf(date)   );
          writeln('      Alternativa 2: DD/MM/',' ',YearOf(date),'   ');
          writeln('      Alternativa 3: DD/MM/20AA      '   );
          Write('      Introduzca nuevamente la fecha      : ');
        end;
      until valido=true ;
    end;
```

UNIDAD 8 ALGORITMO EN LENGUAJE PASCAL

El algoritmo elaborado en lenguaje pascal, compilado en Free Pascal, se entrega en un CD que contiene el programa fuente en lenguaje Pascal (.PAS) y el programa ejecutable (.EXE).

WEBGRAFIA

1.Function existe archivo/fileExists

<http://www.lawebdelprogramador.com/foros/Pascal-Turbo-Pascal/1528229-Ejercicio-de-archivos-Tengo-final-ayuda-Pascal.html>

<http://www.nachocabanes.com/pascal/cupas5/cupas07.php>

2.Algoritmo para validar solo letras y espacios / Nombre y apellido

<http://www.lawebdelprogramador.com/foros/Pascal-Turbo-Pascal/1358383-AYUDA-VALIDAR-SOLO-LETRAS-O-TEXTO-EN-PASCAL.html>

3.Algoritmo para comparar fechas

<http://www.lawebdelprogramador.com/foros/Pascal-Turbo-Pascal/1530854-comparar-fechas.html>

4. Algoritmo para eliminar

<http://www.aprendeaprogramar.com/mod/forum/discuss.php?d=1380>

5.Algoritmo para ordenar registro

<http://www.lawebdelprogramador.com/foros/Pascal-Turbo-Pascal/1357176-URGENTE-ORDENAR-REGISTRO-POR-CEDULA-IDENTIDAD.html#i1357176>

6.Algoritmo para modificar datos

<http://www.lawebdelprogramador.com/foros/Pascal-Turbo-Pascal/1494348-Fichero-secuencial-Modificar-Baja-Ayuda.html>

7.Unidades en FreePascal, web oficial

<http://www.freepascal.org/docs-html/current/rtl/index.html>