

PROGRAMACIÓN BÁSICA – Examen
Evaluación Ordinaria – Continua – GRUPO B
22 de noviembre de 2021

SOLUCIÓN

PARTE 1:

1. **(2 puntos) Implementa** el subprograma Tiene_Sintoma, que dado un paciente y un síntoma determina si dicho paciente tiene ese síntoma.

```
FUNCTION Tiene_Sintoma (P: IN T_Paciente; S: IN T_Sintoma)
RETURN Boolean IS
  --- pre: Los síntomas del paciente están en orden creciente
  --- post: Indicará si el paciente tiene ese síntoma.
  SP : T_Lista_Sintomas := P.Sintomas;
  I : Integer           := 1;
BEGIN
  WHILE I<= SP.Num_Sintomas AND THEN SP.Sintomas(I)<S LOOP
    I:= I+1;
  END LOOP;
  RETURN I<=SP.Num_Sintomas AND THEN SP.Sintomas(I)=S;
END Tiene_Sintoma;
```

2. **(2 puntos) Implementa** el subprograma Padece_Enfermedad_Probable que dada una enfermedad y un paciente indica la probabilidad de que ese paciente tenga esa enfermedad. La probabilidad se obtendrá según la proporción de síntomas de la enfermedad que tiene. Por ejemplo, si una enfermedad se manifiesta con 3 síntomas y el paciente tiene 2 de esos síntomas, entonces su probabilidad será de 2/3 (0,66).

```
FUNCTION Padece_Enfermedad_Probable (P: IN T_Paciente; E: IN T_Enfermedad)
RETURN Float IS
  --- pre: Los síntomas del paciente y de las enfermedades están en orden
  --- creciente
  --- post: Devolverá la probabilidad de que el paciente tenga la enfermedad.

  Cont : Integer := 0;
  SE : T_Lista_Sintomas := E.Sintomas;
BEGIN
  FOR I IN 1..SE.Num_Sintomas LOOP
    IF Tiene_Sintoma(P, SE.Sintomas(I)) THEN
      Cont:= Cont+1;
    END IF;
  END LOOP;
  RETURN Float(Cont)/Float(SE.Num_Sintomas);
END Padece_Enfermedad_Probable ;
```

PARTE 2: Ahora además de los tipos anteriores se usarán estos tipos de datos:

3. **(2,5 puntos) Implementa** Actualiza_Enfermedad que, dada la lista de enfermedades y una enfermedad, la añade a la lista. La lista está ordenada alfabéticamente por nombre de enfermedad y tras ejecutar este subprograma debe seguir en orden. Si la enfermedad ya existe en la lista, entonces modifica solo la lista de sus síntomas. La lista no está llena.

```
PROCEDURE Actualiza_Enfermedad(L: IN OUT T_Lista_Enfermedades;
                                E: T_Enfermedad) IS
  I: Integer:= 1;
BEGIN
  WHILE I<= L.Num_Enfermedades AND THEN
    L.Enfermedades(I).Nombre<E.Nombre LOOP
    I:= I+1;
  END LOOP;
```

PROGRAMACIÓN BÁSICA – Examen

Evaluación Ordinaria – Continua – GRUPO B

22 de noviembre de 2021

```

IF I<=L.Num_Enfermedades AND THEN L.Enfermedades(I).Nombre=E.Nombre THEN
  L.Enfermedades(I).Sintomas:= E.Sintomas;
ELSE
  L.Enfermedades(I+1..L.Num_Enfermedades+1) :=
    L.Enfermedades(I..L.Num_Enfermedades);
  L.Enfermedades(I):= E;
  L.Num_Enfermedades:= L.Num_Enfermedades+1;
END IF;
END Actualiza_Enfermedad;

```

4. (2,5 puntos) Implementa Obtener_Estado_Ambulatorio que, dada la lista de enfermedades y el vector de pacientes, genera una matriz de probabilidades siendo la fila I (la información del paciente I) y en cada columna J se indicará la probabilidad de que padezca la J-ésima enfermedad. Por ejemplo, si el primer paciente del vector VP tuviera 2 síntomas de la primera enfermedad de LE, aparecerá en la posición (1, 1) de la matriz el valor 0,66 (2/3).

La matriz de probabilidades que representa el estado del ambulatorio se implementará con:

```

FUNCTION Obtener_Estado_Pacientes (VP : T_Vector_Pacientes;
                                   LE : T_Lista_Enfermedades) RETURN T_Matriz_Probabilidad IS
  M: T_Matriz_Probabilidad (VP'Range, 1 .. LE.Num_Enfermedades);
BEGIN
  FOR I IN VP'Range LOOP
    FOR J IN 1..LE.Num_Enfermedades LOOP
      M(I, J):= Padece_Enfermedad_Probable(VP(I), LE.Enfermedades(J));
    END LOOP;
  END LOOP;
  RETURN M;
END Obtener_Estado_Pacientes;

```

PARTE 3:

5. (1 punto) Para implementar un programa principal que puede llamar a alguno de los subprogramas anteriores realiza en el mismo programa de pruebas Prueba lo siguiente:

```

PROCEDURE Prueba_Noviembre IS
  -- Declaración de variables solicitadas
  LS0, LS1, LS4 : T_Lista_Sintomas;
  P : T_Paciente;
  E : T_Enfermedad;
BEGIN
  LS0.Num_Sintomas:=0;

  LS1.Sintomas(1):= Dolor_Cabeza;
  LS1.Num_Sintomas:= 1;
  LS4.Sintomas(1..4):= (Dolor_Cabeza, Fiebre, Mareos, Vomitos);
  LS4.Num_Sintomas:=4;

  P.Nombre:= (Others => ' ');
  P.Nombre(1..11):= "Mena, Aritz" ;
  P.Sintomas:= LS4;

  E.Nombre(1..7):= "Gripe A";
  E.Nombre(8..20):= (Others => ' ');
  E.Sintomas:= LS4;
END Prueba_Noviembre;

```