Computabilidad

Apellidos, Nombre:……………………………………………………DNI:…………

1. **(2 puntos)** Indica cuáles serían las funciones semántica **unaria** y **binaria** del siguiente programa while P:

```
begin
 X4 := X1;
 while X4 ≠ X5 do
 begin
  X3 := X4;
  X4 := pred(X4);
  X4 := pred(X4);
 end
 X3 := pred(X3);
 while X3 ≠ X5 do
 begin
  X1 := X2 + X1;
  X3 := pred(X3);
 end
end
```

$$\varphi^{(1)}(x) = x$$

$$\varphi^{(2)}(x,y) = \begin{cases} x & si\ x\ impar\ o\ x = 0 \\ x + y & si\ x\ par\ y\ x > 0 \end{cases}$$

2. **(1.6 puntos)** Dado un programa k-variables Pg cuya función semántica unaria es g(x), completa los **4 huecos** del siguiente programa Programa While de forma que compute la función dada. **Se permiten únicamente las macros de la suma, producto y asignación**:

$$f(x,y,z) = \begin{cases} x \cdot g(y) & si\ z = 0 \\ \perp & si\ z > 0 \end{cases}$$

```
begin
 Xk+1 := X1;
 while X3 ≠ X4 do
 begin
  X3 := X3;
 end
 X1 := X2;
 X2 := 0;
 Pg;
 X1 := X1 * Xk+1;
end
```

3. **(1.6 puntos)** Completa las **4 transiciones** incompletas de tal forma que la siguiente Máquina de Turing M=({0,1}, {q0,q1,q2,q3,q4,q5,q6,qf}, T, q0, {qf}) compute la función ternaria:

$$\varphi^{(3)}(x,y,z) = \begin{cases} z+1 & si\ x \leq y \\ \perp & en\ otro\ caso \end{cases}$$

(q0, 1, 0, D, q1)	(q2, 1, 1, D, q2)	(q4, 1, 1, I, q4)	(q6, 0, 0, H, qf)
(q0, 0, 0, D, q6)	(q2, 0, 0, I, q3)	(q4, 0, 0, I, q5)	(q6, 1, 0, D, q6)
(q1, 1, 1, D, q1)	(q3, 1, 0, I, q4)	(q5, 1, 1, I, q5)	
(q1, 0, 0, D, q2)	(q3, 0, 0, N, q3)	(q5, 0, 0, D, q0)	

4. **(1.5 puntos)** Determina las funciones semánticas **unaria** y **binaria** que computa la siguiente Máquina de Turing M=({0,1}, {q0,q1,q2,q3,q4,q5,q6,q7,qf}, T, q0, {qf}).

(q0, 1, 0, D, q1)	(q3, 0, 0, I, q4)	(q6, 1, 1, I, q5)
(q1, 1, 1, D, q1)	(q4, 1, 0, I, q4)	(q6, 0, 0, H, qf)
(q1, 0, 0, D, q2)	(q4, 0, 0, I, q5)	(q7, 1, 0, D, q7)
(q2, 1, 1, D, q3)	(q5, 1, 0, I, q6)	(q7, 0, 0, H, qf)
(q3, 1, 0, D, q7)	(q5, 0, 0, H, qf)	

$$\varphi^{(1)}(x) = \perp$$

$$\varphi^{(2)}(x,y) = \begin{cases} x+1 & si\ y > 0 \\ x/2 & si\ y = 0 \end{cases}$$

5. Se pretende demostrar que el problema de decidir si un programa while P cumple que su función semántica binaria es $\varphi_P(x,y) = \max\{x,y\}$ es irresoluble.

a) **(1.8 puntos)** Rellena los **5 huecos** que faltan en el proceso de demostración por reducción:

Suponemos que existe un algoritmo A cuya función semántica es:

$$\varphi_A(e) = \begin{cases} 1 & si\ \varphi_e(x,y) = max\{x,y\} \\ 0 & si\ no \end{cases}$$

y definimos la macro A(X) a partir de él.

Hacemos ahora este programa P_d, que nos permitirá más adelante reducir el problema de la parada (se permite usar todas las macros vistas en clase):

```
begin
 X3 := U(c,k);
 if X2 > X1 then
  X1 := X2
end
```

y cuya función semántica es:

$$\varphi_d(x,y) = \begin{cases} max\{x,y\} & si\ P_c\ para\ con\ entrada\ k \\ \perp & si\ no \end{cases}$$

Aplicando el teorema de parametrización, el código d se puede calcular mediante una función f(c,k) total y computable. Dado que es total y computable, se puede definir una macro F(X,Y) que la compute.

A partir de las macros A y F podemos ahora definir el siguiente programa:

```
begin
 X1 := F(X1,X2);
 X1 := A(X1);
end
```

cuya función semántica es:

$$\varphi(c,k) = \begin{cases} 1 & si\ \varphi_{f(c,k)}(x,y) = \varphi_d(x,y) = max\{x,y\} \\ 0 & si\ no \end{cases}$$

Dada la definición de P_d, esta función es equivalente a la función característica del problema de la parada. Por tanto, hemos reducido el problema de la parada al nuestro, pero esto es una contradicción, ya que el problema de la parada es irresoluble, por lo que nuestro problema es irresoluble.

b) **(0.5 puntos)** Demuestra, utilizando el Teorema de Rice, que el problema descrito en el apartado anterior es irresoluble (se permite usar todas las macros vistas en clase).

El problema trata de verificar si un programa P cumple cierta propiedad, en este caso que su función semántica binaria es $\varphi_P(x,y) = \max\{x,y\}$. Dado que la propia propiedad se refiere a la función semántica, ésta es una propiedad semántica. Haría falta comprobar si es una propiedad no-trivial:

```
begin
 if X2 > X1 then
  X1 := X2
end
```

```
begin
 X1 := 1;
end
```

Como podemos construir dos programas, uno que la cumple (izda) y otro que no (dcha), podemos afirmar que es una propiedad no trivial. Por tanto, el problema de verificar si un programa cumple esta propiedad semántica no trivial es irresoluble.

6. **(1 punto, incorrecta resta 0.3)** Dada la función que calcula el punto en un plano como $f(x,y) = ax + by + c$, ¿cuál de las siguientes funciones semánticas sería equivalente a $f(x,y)$ para cualquier valor de a, b y c?

a) $\varphi_{S_3^2(e,x,y)}(a,b,c)$, donde *e* es el código de:

```
begin
 X1 := a * X1;
 X2 := b * X2;
 X1 := X1 + X2;
 X1 := X1 + c;
end
```

b) $\varphi_{S_2^3(e,a,b,c)}(x,y)$, donde *e* es el código de:

```
begin
 X1 := a * X4;
 X2 := b * X5;
 X1 := X1 + X2;
 X1 := X1 + c;
end
```

c) $\varphi_{S_3^2(e,x,y)}(a,b,c)$, donde *e* es el código de:

```
begin
 X1 := X3 * X1;
 X2 := X4 * X2;
 X1 := X1 + X2;
 X1 := X1 + X3;
end
```

d) $\varphi_{S_2^3(e,a,b,c)}(x,y)$, donde *e* es el código de:

```
begin
 X1 := X1 * X4;
 X2 := X2 * X5;
 X1 := X1 + X2;
 X1 := X1 + X3;
end
```

Opción d)