



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Lenguajes de Programacion Examen Parcial IV



■ Edgar Montiel Ledesma
317317794

■ Carlos Daniel Cortes Jimenez
420004846

1. Considera el siguiente programa en el lenguaje While :

```
new z = 0;
while (y > x + 1) do
  (z := z + 1;
   x := x - y)
end
```

- a) Ejecuta el programa en el estado en el que $\sigma(x) = 17$ y $\sigma(y) = 5$ ¿Cual es el estado resultante de la evaluacion?
- b) Da un estado σ tal que si se evalua el programa anterior con dicho estado la evaluacion se ciclaria infinitamente.

2. Extiende el lenguaje While con el operador:

for $x := a1$ to $a2$ do c

esto es:

- a) Modifica la estructura de la maquina W (agregando marcos, estados o transiciones) para evaluar la expresion for.
 - b) Da las reglas de semantica estatica para verificacion de tipos para el nuevo operador for.
 - c) ¿Es posible definir el operador for como azucar sintactica dentro del lenguaje While? justifica tu respuesta.
3. Decimos que dos programas en el lenguaje While son equivalentes ($c1 \equiv_w c2$) si y solo si la ejecucion de ambos programas resulta en el mismo estado, es decir, si para todo estado de las variables σ , $\Diamond \succ \langle c1, \sigma \rangle \rightarrow^* W \Diamond \prec \sigma'$ y $\Diamond \succ \langle c2, \sigma \rangle \rightarrow^* W \Diamond \prec \sigma'$ entonces $c1 \equiv_w c2$.

Con la definicion de equivalencia anterior, demuestra o da un contraejemplo de lo siguiente:

- a) \equiv_w realmente es una relacion de equivalencia. Esto es, demuestra que la relacion \equiv_w es transitiva, reflexiva y simetrica.
- b) $c; \text{skip} \equiv_w c$
- c) $c1; c2 \equiv_w c2; c1$
- d) $c1; (c2; c3) \equiv_w (c1; c2); c3$