Solución Práctica 2 de Modelos Abstractos de Cálculo
Curso 10/11

1. Por supuesto puede haber múltiples soluciones.

1. $L = \{p \mid p \text{ siempre para}\}$.
   Porque si $(p, x) \in H \Rightarrow p(x)\downarrow \Rightarrow h_1(p, x)$ es un programa que siempre devuelve el mismo valor, 2∗resultado donde resultado es la salida de p con entrada $x \Rightarrow h_1(p, x)$ es un programa que siempre para $\Rightarrow h_1(p, x) \in L$;
   si $(p, x) \notin H \Rightarrow p(x)\uparrow \Rightarrow h_1(p, x)$ es un programa que nunca para $\Rightarrow h_1(p, x) \notin L$.
2. $L = \{(q, y) \mid \varphi_q(y) = 7\}$.
   Porque si $(p, x) \in H \Rightarrow p(x)\downarrow \Rightarrow h_2(p, x) = (q, 25)$ donde q es un programa que siempre devuelve $7 \Rightarrow h_2(p, x) \in L$;
   si $(p, x) \notin H \Rightarrow p(x)\uparrow \Rightarrow h_2(p, x) = (q, 25)$ donde q es un programa que nunca para $\Rightarrow h_2(p, x) \notin L$.
3. $L = \{p \mid p \text{ calcula la identidad}\}$.
4. $L = \{p \mid p \text{ devuelve algún impar como resultado}\}$.
5. $L = \{(p, q) \mid p \text{ y } q \text{ hacen lo mismo}\}$. (En clase definimos "hacer lo mismo" como $\varphi_p = \varphi_q$).
6. $L = \{(p, q) \mid \exists x\, \varphi_p(x) = \varphi_q(x)\}$.

2. Se puede utilizar la definición de semidecidible para demostrar que un conjunto es semidecidible.

- $\overline{H} \leq L_1$ mediante la reducción

$f(p, x) =$
```
“Leer N
 constante CP:=p; CX:=x;
 SimularConTiempo(CP, CX, N, ÉXITO);
 Si ÉXITO  entonces
     Devuelve 5
    else
     Devuelve 0.”
```

  $\overline{L_1} = \{x \mid \mathrm{Im}(\varphi_x) \not\subseteq \{0, 1\}\}$. $\overline{L_1}$ es semidecidible porque existe el siguiente programa

```
Leer x
  T:=1; FIN:=falso;
  Mientras que (not FIN) hacer
    Para M:=1 hasta T;
```

```
  SimularConReloj(x,M,T,ÉXITO,RESULTADO);
  Si ÉXITO entonces
    FIN:=FIN OR ((RESULTADO≠ 0) and (RESULTADO≠ 1));
  Fpara;
  T:=T+1;
Fmq;
Devuelve 1.
```

También se puede demostrar con una reducción $\overline{L_1} \leq H$:

$$f(x) = \left(\begin{array}{l} \text{“Leer N} \\ \text{CX:=}x; \\ \quad \text{T := 1; FIN := } falso; \\ \quad \text{Mientras que (not FIN) hacer} \\ \quad\quad \text{Para M:=1 hasta T;} \\ \quad\quad \text{SimularConReloj(CX,M,T,ÉXITO,RESULTADO);} \\ \quad\quad \text{Si ÉXITO entonces} \\ \quad\quad\quad \text{FIN:=FIN OR ((RESULTADO} \neq \text{0) and (RESULTADO} \neq \text{1));} \\ \quad\quad \text{Fpara} \\ \quad\quad \text{T:=T+1} \\ \quad \text{Fmq;} \\ \quad \text{Devuelve 1.”;} \end{array} , 20 \right)$$

Porque si $x \in \overline{L_1} \Rightarrow f(x) = (q, 20)$ donde q es un programa que siempre para $\Rightarrow f(x) = (q, 20) \in H$;
si $x \notin \overline{L_1} \Rightarrow f(x) = (q, 20)$ donde q es un programa que nunca para $\Rightarrow f(x) = (q, 20) \notin H$;

- $\overline{H} \leq L_2$ mediante la reducción

$$f(p,x) = \begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p;\ \text{CX:=}x; \\ \text{SimularConTiempo(CP, CX, N, ÉXITO);} \\ \text{Si ÉXITO entonces} \\ \quad\quad \text{Devuelve 5} \\ \quad \text{else} \\ \quad\quad \text{Devuelve N.”} \end{array}$$

$\overline{L_2} = \{x \,|\, \varphi_x \text{ no es suprayectiva}\}$
$\overline{H} \leq \overline{L_2}$ mediante la reducción

$$g(p,x) = \begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p;\ \text{CX:=}x; \\ \text{Simular(CP, CX, ÉXITO);} \\ \text{Si ÉXITO entonces} \\ \quad\quad \text{Devuelve N.”} \end{array}$$

- (En lugar de la notación P_m, se puede usar directamente m y entender del contexto que m es de tipo programa). L_3 es semidecidible porque existe el siguiente programa

  ```
  Leer m,n
    T:=0; ÉXITO:=falso;
    Mientras que (not ÉXITO) hacer
      T:=T+1;
      SimularConReloj(m,n,T,ÉXITO,RESULTADO);
    Fmq;
    Si (T MOD n)=0 entonces Devuelve 1.
  ```

  $\overline{L_3} = \{m, n \,|\, m(n)\!\uparrow \vee (m(n)\!\downarrow \text{ y tarda tiempo no multiplo de } n\}$.
  $\overline{H} \leq \overline{L_3}$ mediante la reducción

  $$g(p,x) = \left(\begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p\text{; CX:=}x\text{;} \\ \text{Simular(CP, CX, ÉXITO);} \\ \text{Si ÉXITO entonces} \\ \quad \text{Devuelve N.”} \end{array} , 1 \right)$$

- $\overline{H} \leq L_4$ mediante la reducción

  $$f(p,x) = \left(\begin{array}{l} \text{“Leer N} \\ \text{Devuelve N.”} \end{array} , \begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p\text{; CX:=}x\text{;} \\ \text{Simular(CP, CX, ÉXITO);} \\ \text{Si ÉXITO entonces} \\ \quad \text{Devuelve N.”} \end{array} \right)$$

  $\overline{L_4} = \{x, y \,|\, \forall n (x(n)\!\uparrow \vee y(n)\!\downarrow)\}$.
  $\overline{H} \leq \overline{L_4}$ mediante la reducción

  $$g(p,x) = \left(\begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p\text{; CX:=}x\text{;} \\ \text{Simular(CP, CX, ÉXITO);} \\ \text{Si ÉXITO entonces} \\ \quad \text{Devuelve N.”} \end{array} , \begin{array}{l} \text{“Leer N} \\ \text{Mientras que 0=0} \\ \quad \text{M:=3;} \\ \text{Fmq.”} \end{array} \right)$$

- $\overline{H} \leq L_5$ mediante la reducción

  $$f(p,x) = \begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p\text{; CX:=}x\text{;} \\ \text{SimularConTiempo(CP, CX, N, ÉXITO);} \\ \text{Si not ÉXITO entonces} \\ \quad \text{Devuelve N.”} \end{array}$$

$\overline{L_5} = \{x \mid \varphi_x \text{ no es total}\}$.
$\overline{H} \leq \overline{L_5}$ mediante la reducción

$$g(p,x) = \begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p\text{; CX:=}x\text{;} \\ \text{Simular(CP, CX, ÉXITO);} \\ \text{Si ÉXITO entonces} \\ \quad \text{Devuelve N.”} \end{array}$$

- $\overline{H} \leq L_6$ mediante la reducción

$$f(p,x) = \begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p\text{; CX:=}x\text{;} \\ \text{SimularConTiempo(CP, CX, N, ÉXITO);} \\ \text{Si not ÉXITO entonces} \\ \quad \text{Devuelve N.”} \end{array}$$

$\overline{L_6} = \{x \mid \exists n\, x(n)\uparrow \vee (\varphi_x(n) \geq \varphi_x(n+1))\}$.
$\overline{H} \leq \overline{L_6}$ mediante la reducción

$$g(p,x) = \begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p\text{; CX:=}x\text{;} \\ \text{Simular(CP, CX, ÉXITO);} \\ \text{Si ÉXITO entonces} \\ \quad \text{Devuelve N.”} \end{array}$$

- L_7 es semidecidible porque existe el siguiente programa

```
Leer x
  T:=1; FIN:=falso;
  Mientras que (not FIN) hacer
    Para M:=1 hasta T;
    SimularConReloj(x,M,T,ÉXITO,RESULTADO);
    Si ÉXITO entonces
      SimularConReloj(M,x,T,ÉXITO2,RESULTADO2);
      FIN:=FIN OR ÉXITO2;
    Fpara;
    T:=T+1;
  Fmq;
  Devuelve 1.
```

$\overline{L_7} = \{x \mid \forall y (y \notin \mathrm{Dom}(\varphi_x) \vee (x \notin \mathrm{Dom}(\varphi_y))\}$.
$\overline{H} \leq \overline{L_7}$ mediante la reducción

$$g(p,x) = \begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p\text{; CX:=}x\text{;} \\ \text{Simular(CP, CX, ÉXITO);} \\ \text{Si ÉXITO entonces} \\ \quad \text{Devuelve N.”} \end{array}$$

- L_8 es semidecidible porque existe el siguiente programa

```
Leer x, y
  T:=y+1; FIN:=falso;
  Mientras que (not FIN) hacer
    Para M:=y+1 hasta T;
    SimularConReloj(x,M,T,ÉXITO,RESULTADO);
    FIN:=FIN OR (ÉXITO and (RESULTADO mod 2)=0);
    Fpara;
    T:=T+1;
  Fmq;
  Devuelve 1.
```

  $\overline{L_8} = \{x, y \,|\, \forall z > y(x(z)\!\uparrow \vee \varphi_x(z) \text{ es impar})\}$.
  $\overline{H} \leq \overline{L_8}$ mediante la reducción

$$g(p,x) = \left(\begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p\text{; CX:=}x\text{;} \\ \text{Simular(CP, CX, ÉXITO);} \\ \text{Si ÉXITO entonces} \\ \quad \text{Devuelve 2*N.”} \end{array} ,3 \right)$$

- $\overline{H} \leq L_9$ mediante la reducción

$$f(p,x) = \begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p\text{; CX:=}x\text{;} \\ \text{Simular(CP, CX, ÉXITO);} \\ \text{Si ÉXITO entonces} \\ \quad \text{Devuelve N.”} \end{array}$$

  $\overline{L_9} = \{z \,|\, \mathrm{Im}(\varphi_z) \text{ tiene infinitos elementos distintos})\}$.
  $\overline{H} \leq \overline{L_9}$ mediante la reducción

$$g(p,x) = \begin{array}{l} \text{“Leer N} \\ \text{constante CP:=}p\text{; CX:=}x\text{;} \\ \text{SimularConTiempo(CP, CX, N, ÉXITO);} \\ \text{Si not ÉXITO entonces} \\ \quad \text{Devuelve N.”} \end{array}$$

3. Para que se pueda aplicar el Teorema de Rice a un conjunto A este debe ser un conjunto de índices, es decir, un conjunto de programas que cumpla: para cada p, q programas que hacen lo mismo se cumple una de las dos siguientes condiciones

- $p \in A$ y $q \in A$
- $p \notin A$ y $q \notin A$

Podemos aplicar el Teorema de Rice a $L_1, L_2, L_3, L_5, L_6, L_7, L_{10}, L_{11}, L_{14}$.

No podemos aplicarlo a L_4, L_8.

L_9 es el conjunto de índices formado por TODOS los programas, luego es trivialmente decidible. Lo mismo para L_{12}.

L_{13} es el conjunto vacío luego es trivialmente decidible.

4. Según el teorema 3.12 de los apuntes, para todo x se cumple que $\mathrm{Dom}(\varphi_x)$ es semidecidible, por tanto L es el conjunto vacío que es trivialmente decidible.