

Ejercicios propuestos en la clase 4 (capítulo 2)

Las mejores respuestas a este ejercicio entregadas hasta el viernes 5 de octubre tendrán 0,5 puntos en la nota final y serán publicadas en la web de la asignatura

1. Desarrollar el siguiente módulo en Ada que implemente el TAD programa.

```
package simuladorPrograma is

    type cadena is ...; -- tipo cadena de ceros y unos.
    type programa is ...; -- (por ejemplo string) tipo procedimiento en ADA con
    -- un parámetro de entrada de tipo cadena y uno de salida de tipo cadena.

    procedure ejecutar(p:in programa; x:in cadena; y:out cadena);
    -- Este procedimiento ejecuta p con entrada x. Si para el resultado final
    -- está en y.

    procedure simular(p:in programa; x:in cadena; éxito:out boolean);
    -- Este procedimiento ejecuta p con entrada x. Si para éxito=true.

    procedure simularConReloj(p:in programa; x:in cadena; t:in integer;
    éxito:out boolean);
    -- Este procedimiento ejecuta t pasos de p con entrada x. Si para en t
    -- pasos o menos éxito=true, si no éxito=false.

    procedure leeDeFichero(f:in filetype; p:out programa);
    procedure escribeAFichero(p:in programa; f:in filetype);

    -- Los siguientes procedimientos hacen el cambio de tipo.
    procedure programaAcadena(p:in programa; cp:out cadena);
    procedure programaAnumero(p:in programa; np:out natural);
    procedure cadenaAprograma(cad:in cadena; pcad:out programa);
    procedure numeroAprograma(num:in natural; pnum:out programa);

    procedure calculaSelf(p: in programa; sp:out programa);
    -- Este procedimiento devuelve un programa sp que para cualquier entrada
    -- hace lo mismo que p(p).
    -- OPCIONAL, no es necesario para el 0,5

end simuladorPrograma;
```

2. Terminar lo que falte y probar que el resultado del siguiente programa es un autoreplicante.
OPCIONAL, no es necesario para el 0,5.

```
with simuladorPrograma;
use simuladorPrograma;
procedure paraAutoReplicar(x:in cadena; resultado:out programa) is
    -- El resultado de este programa es un autoreplicante
    var p:programa;
    begin
        p:= ``with simuladorPrograma;\n use simuladorPrograma;\n procedure uno(t:in
        cadena; s:out cadena) is \n var a,b:programa;\n begin\n cadenaAprograma(t,a);\n
        calculaSelf(a,b);\n programaAcadena(b,s);\n end uno;'';

        calculaSelf(p,resultado);
    end paraAutoReplicar;
```