



# 4º Ingeniería Informática

## II26 Procesadores de lenguaje

Control de la práctica de microcalc (13 de diciembre de 2008)

### INSTRUCCIONES:

- La duración del examen es de dos horas.
- Antes de empezar, asegúrate de que el usuario con el que estás trabajando coincide con el del recuadro del final de esta página. Rellena el recuadro con tus datos.
- Crea en el *home* de tu usuario un fichero llamado `datos` que contenga tres líneas: la primera con tu DNI, la segunda con tu nombre y la tercera con tus apellidos. Por ejemplo:

```
007F
Bond
James
```
- Al introducir el USB, debería montarse automáticamente en el directorio `/media/disk`, si no tiene ninguna partición, o en `/media/disk1`, si los datos están en la partición 1. Si fuera necesario, puedes montarlo y desmontarlo mediante `mount` y `umount` sobre el directorio apropiado.
- Cuando termines el examen, entrérganos esta hoja.
- A continuación tienes dos extensiones de `microcalc` *independientes entre sí*. Crea en el directorio *home* de tu usuario dos directorios `ext1` y `ext2`. Implementa en cada uno de ellos la correspondiente extensión de tu intérprete. En ambos casos, debes tener un fichero ejecutable que se llame `microcalc`. No es necesario que implementes ninguna de las opciones `-a`, `-l` o `-s`.

### ext1. Operador de conversión (0,3 puntos)

Modifica tu implementación de `microcalc` para que acepte el nuevo operador de conversión (`<<<>>`). Este operador es sintácticamente similar a los paréntesis y al operador barra: rodea una expresión y le da la máxima prioridad. El efecto del operador depende del tipo de la expresión. Si la expresión es entera, el resultado es una cadena con la representación en base 10 del número. Por ejemplo, el resultado de `<<3*3>>+"1"` es la cadena `91`. Cuando la expresión es de tipo cadena, el resultado es el entero que resulta de interpretar la cadena como una secuencia de dígitos en base 10. Por ejemplo, el resultado de `<<3*"2">>+1` es el entero `223`. Si la cadena no cumple las reglas de `microcalc` para los enteros (es decir, si no pertenece al lenguaje `0|[1-9][0-9]*`), se produce un error de ejecución.

Tendrás que modificar la gramática. Te sugerimos que añadas la siguiente producción:

$$\langle \text{Factor} \rangle \rightarrow \text{aconv} \langle \text{Expresión} \rangle \text{cconv}$$

Donde `aconv` y `cconv` son las categorías léxicas que representan la apertura y cierre del operador de conversión.

Recuerda que puedes convertir una cadena que contenga dígitos en el correspondiente entero mediante `int` y un entero en cadena mediante `str`.

### ext2. Comentarios (0,2 puntos)

Modifica tu implementación de `microcalc` para que acepte comentarios. Estos comienzan por la secuencia `(*` y terminan en la primera secuencia `*)` que le siga. En su interior pueden contener cualquier carácter excepto el fin de línea.

DNI:  
Nombre:  
Usuario: `exam300`