

Compiladors: Examen de laboratori de Asl

20 de març de 2015

Duració de l'examen: fins les 17:00h (no es podrà fer cap entrega més enllà d'aquesta hora).

Cal enviar l'interpret complet en un fitxer `tgz` i fer l'entrega a través del Racó.

Al Racó trobareu un fitxer `examen.tgz` que conté:

- Aquest enunciat
- Interpret ASL original
- Jocs de proves per validar els exercicis proposats a l'examen
- Un script `avalua.sh` que executa l'ASL sobre els jocs de proves
- Un script `empaqueta.sh` que crea un fitxer `entrega.tgz` per pujar al Racó

1 Operador condicional ternari (5 punts)

Afegir una nova forma d'expressió a l'interpret de Asl que serà l'operador condicional ternari:
 $x > y ? x / 2 : (x+1)/2$.

El valor d'aquesta expressió es calcula evaluant en primer lloc el primer operand, la condició $x > y$: si aquesta és certa, el valor de tota l'expressió serà el del segon operand, la expressió $x/2$, i si la condició és falsa, serà el valor del tercer operand, $(x+1)/2$.

La prioritat de l'operador condicional ternari és inferior a la de tots els altres operadors.

```
func main()
  a = 5;
  write a + (1 > 2 ? 103 : 104);    // escriu "109"
  write "%n";
  write a + 1 > 2 ? 103 : 104;      // escriu "103"
  write "%n";
  write (1 < 2 ? 3*3 : 4) * 5;      // escriu "45"
  write "%n";
  write 1 < 2 ? 3*3 : 4 * 5;        // escriu "9"
  write "%n";
endfunc
```

Jocs de proves: `cond_tern_1.asl` (que veieu a sobre), `cond_tern_2.asl` i `cond_tern_3.asl`.

Modificar l'interpret per a introduir el nou operador.
Només cal modificar els fitxers `Asl.g` i `Interp.java`.

2 Tipus *string* (5 punts)

Afegir el tipus `string` al llenguatge Asl. Des del punt de vista semàntic el podem veure com un altre tipus “bàsic”. Un `string` no té una mida màxima, i ocupa en memòria l’espai necessari per guardar la cadena de caràcters que té assignada.

Ara podem tenir expressions que siguin de tipus `string`. També s’han de poder assignar strings i concatenar-los (sobrecarregant l’operador `+`).

```
func main()
  s1 = "hola";
  write s1;                      // escriu "hola"
  write "%n";
  write s1 + " que tal?";        // escriu "hola que tal?"
  write "%n";
  s1 = s1 + " adeu!";
  write s1;                      // escriu "hola adeu!"
  write "%n";
endfunc
```

Jocs de proves: `string_1.asl` (que veieu a sobre) i `string_2.asl`.

Modificar l’interpret per a introduir el nou tipus amb les seves operacions.

Només cal modificar els fitxers `Asl.g`, `Data.java` i `Interp.java`.