

Compiladors: Examen parcial de laboratori.

8 d'abril de 2026

ATENCIÓ: Al Racó trobareu els jocs de proves i codi necessari per a fer l'examen. El paquet conté versions modificades dels mòduls `common/SemErrors.*` i `common/TypesMgr.*` amb els mètodes necessaris per donar els nous errors o gestionar els nous tipus que apareixen als exercicis de l'examen. **ABANS DE COMENÇAR A FER RES**, llegiu les instruccions del final de l'enunciat per veure com descarregar-lo i instal·lar-lo.

ATENCIÓ: Cal entregar l'examen en un fitxer `.tgz` pujat al Racó. **Llegiu les instruccions del final de l'enunciat per veure com generar-lo.**

PUNTUACIÓ: Els tres primers punts de la nota de laboratori s'obtenen amb els jocs de proves de la pràctica base. La resta s'obtenen superant els jocs de proves específics de l'examen. La correcció és **automàtica**, a través dels jocs de proves d'aquest enunciat, més un conjunt addicional de jocs de proves privats.

IMPORTANT: L'examen consta de tres exercicis. Es recomana fer cada exercici incrementalment, resolent cada joc de proves abans de passar al següent. **ATENCIÓ:** El primer exercici és necessari per fer qualsevol dels altres dos, que són independents entre ells.

Strings i Maps

Volem afegir a l'ASL el tipus bàsic `string`, i el tipus `map` (a l'estil dels maps de C++). Podrem doncs escriure programes com aquest::

```
1 func main()
2     var i,j: int
3     var s,t:string
4     var m1: map<string,bool>
5     var m2,m3: map<int,string>
6
7     m2[i+1] = "hello";
8     m1[s] = i>=j or m2[j]=="bye";
9
10    while m1[s] do
11        i = 2*j - 1;
12        m1[m2[3*i]] = not m1[s+t] or m1["hey"];
13        if "grf" in s then
14            i = index("f", s) * 2;
15        else
16            i = t.length - 1;
17        endif
18    endwhile
19
20    m3 = m2;
21 endfunc
```

1 Tipus bàsic string

ATENCIO: Aquest exercici és imprescindible per poder fer qualsevol dels dos exercicis següents.

Joc de proves 1 (0.5 punts). El primer a fer és afegir el tipus bàsic `string`, i gestionar assignacions i comparacions d'aquest tipus.

Afegeu a la gramàtica el tipus bàsic `string`. Recordeu que ja existeix un token `STRING` que en realitat correspon a un `STRINGVAL` i no al nom del tipus.

Elimineu de la gramàtica la regla `statement:WRITE STRING`, ja que ara les expressions inclouran literals de strings per tant ja entraran per `WRITE expr`.

Adapteu el *SymbolsVisitor* per tal que les variables string es declarin adequadament amb el nou tipus.

Adapteu el *TypeCheckVisitor* per tal que els literals de strings es decorin adequadament amb el nou tipus.

Amb això haurieu de superar el primer joc de proves:

El codi següent:

```
1 func main()
2   var i,j: int
3   var s1,s2: string
4   var A: array[10] of string
5   var B: array[10] of char
6
7   s1 = s2;
8   s1 = "hello";
9   i = s1 * 2;
10  s2 = i+1;
11  s1 = 'q';
12  read s2;
13  if s2 == "q" then
14    write s1;
15    while s1 > "bye" and s1 != 'q' do
16      A[s1] = s2;
17      read s1;
18    endwhile
19  else
20    A[i] = 13;
21  endif
22  A["13"] = 'i';
23  B[j+1] = A[j+1];
24  A = B;
25  A[j-1] = B[j-1];
26 endfunc
```

genera els errors:

```
Line 9:9 error: Operator '*' with incompatible types.
Line 10:5 error: Assignment with incompatible types.
Line 16:9 error: Array access with non integer index.
Line 20:10 error: Assignment with incompatible types.
Line 22:4 error: Array access with non integer index.
Line 23:9 error: Assignment with incompatible types.
Line 24:4 error: Assignment with incompatible types.
```

2 Operacions sobre strings

Joc de proves 2 (0.5 punts). A continuació afegirem a la gramàtica diverses operacions sobre strings:

- Cerca de substrings amb l'operador 'in' (e.g. if "hey" in s1 then ...)
- Funció *built-in* 'index' (e.g. i = index("hey", s1);)
- Mètode *built-in* 'length' (e.g. i = s1.length;)

Per superar aquest joc de proves només és necessari adaptar la gramàtica per tal que reconegui les operacions anteriors:

Així, el segon joc de proves:

```
1 func addA(s : string) : string
2   if s.length == 0 then
3     return "null";
4   else
5     if s.length < 10 then
6       return "1";
7     else
8       return 'X';
9     endif
10  endif
11 endfunc
12
13 func main()
14   var i,j: int
15   var s1,s2: string
16   var A: array[10] of string
17   var B: array[10] of char
18   var c: char
19
20   s1 = x + 1;
21   s1 = c;
22   c = s2;
23
24   if s1 in s2
25     and 'a' in addA('b') then
26     write 1;
27   else
28     if s2 in s1 == true then
29       write 2;
30     endif
31     n = 'a' in s2 and c;
32   endif
33
34   if index(s2, s1) >= 0 then
35     while index('a', s2) + 1 do
36       s2 = s2 / 2;
37     endwhile
38   endif
39   i = index(s1, "a") - c;
40
41   j = s2.length;
42   j = "3".length + A;
43   j = addA(14);
44
45   A[j] = 'c';
46   A["33"] = d + 1;
47
48 endfunc
```

genera els errors:

```
Line 20:5 error: Assignment with incompatible types.
Line 20:7 error: Identifier 'x' is undeclared.
Line 22:4 error: Assignment with incompatible types.
Line 31:4 error: Identifier 'n' is undeclared.
Line 31:18 error: Operator 'and' with incompatible types.
Line 35:5 error: Instruction 'while' requires a boolean condition.
Line 36:12 error: Assignment with incompatible types.
Line 36:17 error: Operator '/' with incompatible types.
Line 39:21 error: Operator '-' with incompatible types.
Line 42:17 error: Operator '+' with incompatible types.
Line 43:4 error: Assignment with incompatible types.
Line 43:11 error: Parameter #1 with incompatible types in call to 'addA'.
Line 46:4 error: Array access with non integer index.
Line 46:10 error: Assignment with incompatible types.
Line 46:12 error: Identifier 'd' is undeclared.
```

Joc de proves 3 (1 punts). Seguidament farem la comprovació de tipus de les noves operacions sobre strings.

- Cerca de substrings amb l'operador `'in'`: Afegir comprovacions de tipus del nou operador. El resultat serà booleà. S'admet coerció de caràcters a strings pel primer operand, però no pel segon. La prioritat d'aquest operador és superior a la dels operadors relacionals, però inferior a la dels aritmètics.
- Funció *built-in* `'index'`: Afegir comprovacions de tipus. El resultat és enter. S'admet coerció de caràcters a strings pel primer operand, però no pel segon.
- Mètode *built-in* `'length'`: Afegir comprovacions de tipus. El resultat és enter. No s'admet coerció de caràcters.

I també afegirem concatenació de strings amb l'operador `'+'`: No cal modificar la gramàtica, però si el TypeCheck per gestionar la sobrecàrrega de la suma. Cal ampliar les comprovacions per admetre que els dos operands puguin ser strings, i el resultat sigui string en aquest cas. També cal admetre coercions de caràcter a string (és a dir, si un dels operands es caràcter, es tracta com si fos string).

ATENCIÓ: Per fer aquesta darrera modificació, distingeix el cas en que l'operador es PLUS i almenys un dels dos operands és string o caràcter, i tracteu-lo separatament de la resta de casos (és a dir, del tractament existent dels operadors aritmètics).

Així, el joc de proves:

```

1 func addA(s : string) : string
2   if s.length == 0 then
3     return "null";
4   else
5     if s.length < 10 then
6       return "1";
7     else
8       return 'X';
9     endif
10  endif
11 endfunc
12
13 func main()
14   var i,j: int
15   var s1,s2: string
16   var A: array[10] of string
17   var B: array[10] of char
18   var c: char
19
20   s1 = s1 + s2;
21   s1 = s1 + 3.0;
22   s1 = s1 / 3.0;
23   s1 = x + s2;
24   s1 = (j > 1) + 'a';
25   s1 = j > 1 + 'a';
26   s1 = 'b' + 'y' + 'e';
27   s1 = 'b' + y + 'e';
28
29   if s1 in s2 and
30     'a' in s1 + addA('b') then
31     write 1;
32   else
33     if s2 in 'b' == true then
34       write 2;
35     endif
36     n = 'a' in s2 > c;
37   endif
38
39   if index(s2, s1) >= 0 then
40     while index('a', s2) > c do
41       s2 = s2 / 2;
42     endwhile
43   endif
44   i = index(s1, 'a') - c;
45
46   j = s2.length * s1;
47   j = '3'.length + A;
48   j = addA(14);
49
50   A[j] = 'c';
51   A["33"] = (A[7] + 1) * A[c];
52
53 endfunc

```

genera els errors:

```

Line 21:10 error: Operator '+' with incompatible types.
Line 22:5 error: Assignment with incompatible types.
Line 22:10 error: Operator '/' with incompatible types.
Line 23:7 error: Identifier 'x' is undeclared.
Line 24:15 error: Operator '+' with incompatible types.
Line 25:5 error: Assignment with incompatible types.
Line 25:9 error: Operator '>' with incompatible types.
Line 25:13 error: Operator '+' with incompatible types.
Line 27:14 error: Identifier 'y' is undeclared.
Line 33:10 error: Operator 'in' with incompatible types.
Line 36:4 error: Identifier 'n' is undeclared.
Line 36:18 error: Operator '>' with incompatible types.
Line 40:26 error: Operator '>' with incompatible types.
Line 41:12 error: Assignment with incompatible types.
Line 41:17 error: Operator '/' with incompatible types.
Line 44:6 error: Operator 'index' with incompatible types.
Line 44:21 error: Operator '-' with incompatible types.
Line 46:16 error: Operator '*' with incompatible types.
Line 47:9 error: Operator '.' with incompatible types.
Line 47:17 error: Operator '+' with incompatible types.
Line 48:4 error: Assignment with incompatible types.
Line 48:11 error: Parameter #1 with incompatible types in call to 'addA'.
Line 51:4 error: Array access with non integer index.
Line 51:10 error: Assignment with incompatible types.
Line 51:18 error: Operator '+' with incompatible types.
Line 51:23 error: Operator '*' with incompatible types.
Line 51:27 error: Array access with non integer index.

```

3 Estructures de tipus map

Aquest exercici consisteix en afegir a ASL estructures de diccionaris, semblants als maps de C++

Joc de proves 4 (1.5 punts). El primer que cal fer es modificar la gramàtica per permetre la declaració de maps. La sintàxi de la declaració és `map<t1,t2>` on `t1` i `t2` són tipus bàsics (incloent `string`).

Haureu de modificar el *SymbolsVisitor* perquè crei el tipus adequat. El mòdul *TypesMgr* modificat ja ofereix mètodes per crear i gestionar els tipus `map`.

L'operació `[]` d'accés a arrays ara està sobrecarregada. Haureu de modificar el *TypeCheckVisitor* per:

- Si la base és un map:
 - Comprovar que l'índex és del tipus de la clau.
 - Decorar el resultat amb el tipus dels valors del map.
- Si la base no és un map (i.e., és un array, o alguna altra cosa):
 - Aplicar el mateix tractament d'arrays que es feia fins ara.

El mòdul *SemErrors* modificat ja ofereix el mètode per donar el nou error relatiu a maps: *invalidKeyTypeInMapAccess*.

En aquest joc de proves només cal modificar els accessos a arrays en expressions. Les *left_expr* es tracten al següent joc de proves.

El següent joc de proves:

```
1 func q(s: string, x: int, m: map<string,bool>) : string
2   var q: array[3] of string
3   s = q[x];
4   q[0] = "156";
5   return q[s] or (m[s] and s > "A");
6 endfunc
7
8 func main()
9   var i,j: int
10  var s,t: string
11  var m1: map<string,bool>
12  var m2,m3: map<int,string>
13  var A: array[17] of bool
14  var B: array[17] of string
15
16  m2 = q("hello", i>2, m1);
17  s = i>=j or m2[j]=="bye";
18  B[s] = B[ m2[j] or not m1[t] ];
19
20  while m1[s] do
21    i = 2*j - 1;
22    s[3*i] = not m1["hey"];
23    if "grf" then
24      i = A[i*2];
25    else
26      t = m2[i-3.4] * 4;
27    endif
28    j = C[i] / 2;
29    t = A[j+1] or s[i] > 3;
30    s = m3[B[0]] > "A" and m1[0];
31  endwhile
32
33  j = m3[i+3.4];
34  m3 = m2;
35 endfunc
```

ha de produir els errors:

Line 5:3 error: Return with incompatible type.
Line 5:12 error: Array access with non integer index.
Line 5:15 error: Operator 'or' with incompatible types.
Line 16:6 error: Assignment with incompatible types.
Line 16:19 error: Parameter #2 with incompatible types in call to 'q'.
Line 17:5 error: Assignment with incompatible types.
Line 18:5 error: Array access with non integer index.
Line 18:13 error: Array access with non integer index.
Line 18:19 error: Operator 'or' with incompatible types.
Line 22:7 error: Array access to a non array operand.
Line 23:7 error: Instruction 'if' requires a boolean condition.
Line 24:12 error: Assignment with incompatible types.
Line 26:12 error: Assignment with incompatible types.
Line 26:17 error: Invalid key type in map access.
Line 26:24 error: Operator '*' with incompatible types.
Line 28:11 error: Identifier 'C' is undeclared.
Line 29:9 error: Assignment with incompatible types.
Line 29:21 error: Array access to a non array operand.
Line 30:9 error: Assignment with incompatible types.
Line 30:14 error: Invalid key type in map access.
Line 30:33 error: Invalid key type in map access.
Line 33:5 error: Assignment with incompatible types.
Line 33:10 error: Invalid key type in map access.

Joc de proves 5 (1 punts). A continuació farem les mateixes modificacions a les regles de *left_expr*.

Això ens permetrà superar el joc de proves següent:

Aquest joc de proves:

```
1 func q(s: string, x: int, m: map<string,bool>) : string
2   var q: array [3] of string
3   m[x] = x;
4   q[0] = "156";
5   return m[q[x]] or (m[s] > "A");
6 endfunc
7
8 func main ()
9   var i,j: int
10  var x,y: float
11  var s,t: string
12  var m1: map<string, bool>
13  var m2,m3: map<int, string>
14  var A: array [17] of bool
15  var B: array [17] of string
16
17  m2[0] = q("hello", i+j, m2[s]) * 4;
18  B[s] = A[ m2[j] ] or not m1[t];
19
20  while m1[s] < m1["ZZ"] do
21    i = 2*j - 1;
22    s[3*i] = not m1["hey"];
23    if "grf" then
24      write not A[2*i];
25    else
26      m2[i-3.4] = t;
27    endif
28    m3[B[0]] = s > "A" and m1[0];
29    m1[B[0]] = m3[99] > "bye";
30  endwhile
31
32  C[x] = m2[128];
33 endfunc
```

produirà els errors:

Line 3:4 error: Invalid key type in map access.
Line 3:7 error: Assignment with incompatible types.
Line 5:2 error: Return with incompatible type.
Line 5:26 error: Operator '>' with incompatible types.
Line 17:8 error: Assignment with incompatible types.
Line 17:26 error: Parameter #3 with incompatible types in call to 'q'.
Line 17:29 error: Invalid key type in map access.
Line 17:33 error: Operator '*' with incompatible types.
Line 18:4 error: Array access with non integer index.
Line 18:7 error: Assignment with incompatible types.
Line 18:12 error: Array access with non integer index.
Line 20:14 error: Operator '<' with incompatible types.
Line 22:4 error: Array access to a non array operand.
Line 23:4 error: Instruction 'if' requires a boolean condition.
Line 26:9 error: Invalid key type in map access.
Line 28:7 error: Invalid key type in map access.
Line 28:13 error: Assignment with incompatible types.
Line 28:30 error: Invalid key type in map access.
Line 32:2 error: Identifier 'C' is undeclared.
Line 32:4 error: Array access with non integer index.

Joc de proves 6 (0.5 punts). El següent pas es admetre coercions en els accessos a maps:

- Si la clau d'un map és de tipus string, s'admet un accés amb una clau caràcter.
- Si la clau d'un map és de tipus real, s'admet un accés amb una clau de tipus enter.
- Si els valors d'un map són de tipus string, s'hi poden assignar o comparar caràcters. Si són valors reals, s'hi poden assignar (o operar amb) enters.

Al mòdul *TypesMgr*, els mètodes *equalTypes*, *compatibleTypes*, i *copyableTypes* han estat convenientment adaptats.

Amb això resoldrem el següent joc de proves.

Aquest joc de proves:

```
1 func q(c: char, x: int, m: map<string,bool>) : char
2   var p: array [3] of char
3   m['c'] = x > 0;
4   return m[p[3*c]] or (m["in q"] > true);
5 endfunc
6
7 func main ()
8   var i,j: int
9   var x: float
10  var c: char
11  var s: string
12  var m1: map<string, bool>
13  var m2,m3: map<float, char>
14  var A: array [17] of char
15
16  m2[i-1] = q("in main", x+y, m1);
17  A[i] = m1[ m2[3.14] ] or not m1["hello"];
18
19  while A[not m1[s]] do
20    s[3*i] = not m1["hey"];
21    if i >= x/2 then
22      write not A[2*i];
23    else
24      m2[i-3.4] = t;
25    endif
26    m3[A[0]] = s > "A" and m1[0];
27    m1[A[0]] = m3[99] > 'a';
28  endwhile
29
30  C[x] = m2[128];
31 endfunc
```

genera els errors:

```
Line 4:2 error: Return with incompatible type.
Line 4:14 error: Operator '*' with incompatible types.
Line 4:33 error: Operator '>' with incompatible types.
Line 16:14 error: Parameter #1 with incompatible types in call to 'q'.
Line 16:25 error: Parameter #2 with incompatible types in call to 'q'.
Line 16:27 error: Identifier 'y' is undeclared.
Line 17:7 error: Assignment with incompatible types.
Line 19:2 error: Instruction 'while' requires a boolean condition.
Line 19:10 error: Array access with non integer index.
Line 20:4 error: Array access to a non array operand.
Line 22:12 error: Operator 'not' with incompatible types.
Line 24:18 error: Identifier 't' is undeclared.
Line 26:7 error: Invalid key type in map access.
Line 26:13 error: Assignment with incompatible types.
Line 26:30 error: Invalid key type in map access.
Line 30:2 error: Identifier 'C' is undeclared.
Line 30:4 error: Array access with non integer index.
```

Joc de proves 7 (0.5 punts). El darrer exercici consistirà en afegir una instrucció d'iteració sobre els elements d'un map.

La instrucció tindrà la sintaxi

```
for k,v in m do
  ...
endfor
```

Caldrà modificar la gramàtica per afegir la nova instrucció, i modificar el *TypeCheckManager* per comprovar que la variable del map (m a l'exemple) està declarada i es un map, i que les variables per a la clau i el valor (k, v a l'exemple) estan declarades i son dels tipus adequats per al map, sense admetre coercions de tipus.

Passarem aquest joc de proves:

```
1 func q(c: char, b: bool, m: map<string,bool>) : bool
2   var p: array [3] of char
3   var s: string
4   for s, b in m do
5     m[s] = not s or b;
6   endfor
7   return m[p[3*c]] or (m["in q"] > true);
8 endfunc
9
10 func main()
11   var i,j: int
12   var x,y: float
13   var c: char
14   var s: string
15   var m1: map<string, bool>
16   var m2: map<float, int>
17   var m3: map<int, float>
18   var A: array [17] of char
19
20   for x,i in m2 do
21     for j,y in m3 do
22       m3[j] = x;
23       m2[y] = i;
24     endfor
25   endfor
26   if q('a', false, m1["hello"]) then
27     for i,c in m2 do
28       write i; write '\n'; write c; write '\n';
29     endfor
30   else
31     for x,i in m3 do m3[x] = i+1; endfor
32     for i,x in m3 do m2[i] = x+1; endfor
33   endif
34   for i,s in m2 do m1[s] = i; endfor
35   if m1['a'] or m2['a'] then
36     for z1,s in m3 do
37       for x,c in A do
38         A[2*x] = m1[s];
39         if m1[c] then
40           m3[z1] = m2[c];
41         endif
42       endfor
43     endfor
44   endif
45
46 endfunc
```

generant els errors:

```
Line 5:11 error: Operator 'not' with incompatible types.
Line 7:14 error: Operator '*' with incompatible types.
Line 7:33 error: Operator '>' with incompatible types.
Line 26:19 error: Parameter #3 with incompatible types in call to 'q'.
Line 27:8 error: For statement applied to incompatible key variable 'i'.
Line 27:10 error: For statement applied to incompatible value variable 'c'.
Line 31:8 error: For statement applied to incompatible key variable 'x'.
Line 31:10 error: For statement applied to incompatible value variable 'i'.
Line 31:24 error: Invalid key type in map access.
Line 32:27 error: Assignment with incompatible types.
Line 34:13 error: Identifier 'mz' is undeclared.
Line 34:25 error: Assignment with incompatible types.
Line 35:13 error: Operator 'or' with incompatible types.
Line 35:19 error: Invalid key type in map access.
Line 36:8 error: Identifier 'z1' is undeclared.
Line 36:11 error: For statement applied to incompatible value variable 's'.
Line 37:17 error: For statement applied to a non-map 'A'.
Line 38:10 error: Array access with non integer index.
Line 38:15 error: Assignment with incompatible types.
Line 40:6 error: Identifier 'z1' is undeclared.
Line 40:15 error: Invalid key type in map access.
```

Informació important

FITXERS PER A L'EXAMEN: Al Racó (`examens.fib.upc.edu`) trobareu un fitxer `examen.tgz` amb el següent contingut:

- `parcial-lab-CL-2026.pdf`: Aquest document, amb l'enunciat i les instruccions.
- `jps`: Subdirectori amb jocs de proves (`jp_chkt_XX.asl`), i la seva corresponent sortida esperada (`jp_chkt_XX.err`).
- `common`: Subdirectori amb els mòduls auxiliars `SemErrors` i `TypesMgr` ampliat amb el codi necessari per a l'examen.
- `evalua.sh`: Script que executa tots els jocs de proves i diu si se superen o no.
- `empaqueta.sh`: Script que crea un fitxer `examen-nom.cognom.tgz` amb la vostra solució. Aquest és el fitxer que cal pujar al Racó.

PASSOS A SEGUIR:

- Feu una còpia de les carpetes `asl` i `common` de la vostra pràctica a un nou directori `examen`.

```
mkdir examen
cp -r practica/asl practica/common examen/
```

- Canvieu al nou directori `examen`, i descomprimiu-hi el fitxer `examen.tgz` del Racó:

```
cd examen
tar -xzf examen.tgz
```

Això extraurà el contingut del paquet, **afegint** al vostre directori `examen` els fitxers llistats anteriorment.

IMPORTANT: Feu-ho en l'ordre especificat (primer una còpia de la vostra pràctica i després descomprimir el `.tgz`). Fer-ho en l'ordre invers causarà que us falti codi necessari a `common` i que els JPs no siguin els adequats.

- Trebal·leu normalment a la carpeta `examen/asl`.

```
cd asl
make antlr
make -j8
...
```

(Si la compilació és lenta per sobrecàrrega del servidor de disc, podeu executar l'script `fast-make.sh`, que fa una còpia temporal i compila al disc local)

- Per veure les diferències entre la sortida del vostre `asl` i la sortida esperada en un joc de proves concret de type check, podeu fer:

```
./asl ../jps/jp_chkt_XX.asl | diff -y - ../jps/jp_chkt_XX.err
```

(Podeu ignorar la línia “`There are semantic errors: no code generated`” que genera el `main`)

- Per executar tots els jocs de proves i veure si els passeu, executeu `../evalua.sh`.
- Executeu `../empaqueta.sh` per crear el fitxer d'entrega `../examen-USERNAME.tgz` que cal pujar al Racó.
Els paquets creats sense usar aquest script seran qualificats com **NO PRESENTAT**.