

Compiladors (CL) GEI (2017-18)

Pràctica 3: Generació de codi intermedi

Objectiu

Afegir generació de codi intermedi (codi de tres adreces) al compilador iniciat a les Pràctiques 1 i 2.

Aprenentatges

Generació de codi intermedi mitjançant back-patching.

Llenguatge font

Descrit a l'enunciat de la Pràctica 2, amb aquestes modificacions:

- La generació de codi de tres adreces per variables booleanes és opcional.
- La generació de codi de tres adreces per cadenes és opcional.
- Es poden afegir, de forma opcional, altres tipus de dades o sentències de flux de control, e.g. taules, registres, condicionals tipus switch, iteratives tipus do-while, instruccions break i continue en iteratives, iteratives sense fi, procediments i funcions, etc.

Codi de tres adreces

En el document *C3A.pdf* es descriu el llenguatge intermedi de codi de tres adreces de forma completament general. Per fer aquesta pràctica no es necessita tot, sinó només un subconjunt d'aquest llenguatge. Els detalls a tenir en compte són:

- El programa és pot considerar directament com format per un únic subprograma.
- Els tipus de dades Long i Double no s'utilitzen.
- Com no hi ha àmbits, totes les variables es poden considerar com a locals.
- La instrucció START no és necessària.
- Els únics subprogrames que es necessiten són els procediments predefinitos PUT.
- Com no hi ha taules ni registres en el llenguatge font, no cal utilitzar consultes ni assignacions desplaçades.
- Com no hi ha apuntadors, no cal utilitzar les proposicions de codi de tres adreces corresponents.

Sortides

En mode calculadora:

- Les assignacions mostraran el nom, el seu tipus i el seu valor.
- Les expressions mostraran el seu tipus i el seu valor.

En mode programa:

- Generació del codi de tres adreces, utilitzant back-patching per les expressions booleanes i pel flux de control.
- Les sentències que són només expressions (i.e. sense assignació) indiquen que es vol imprimir el seu valor per pantalla (i.e. utilitzant els procediments predefinits PUT).

Altres sortides:

- Convé anar guardant en un arxiu de log cada producció de la gramàtica reconeguda, per així controlar el progrés i correcció de les anàlisis lèxica i sintàctica. També es pot guardar al mateix arxiu qualsevol altre missatge informatiu.
- Quan hi ha un error al codi font s'ha d'emetre un missatge d'error indicant la posició actual a l'arxiu font, i si l'error és lèxic, sintàctic o semàntic. Mentre més informació es doni dels errors, millor.

Exemple en mode programa

Entrada:

```
calc off

// nested sentences
total := 0.0
i := 1
while (total < 1000.0) do
  if (i = 1) then
    total := total * 2
  else
    total := total - 1
  fi
  i := -i
  if (total > 666.6 or total = 500.0) then
    i := 1
  fi
done
total
```

Sortida (pot haver-hi variacions):

```
1: $t01 := 0.0
2: total := $t01
3: $t02 := 1
4: i := $t02

5: IF total LTF 1000.0 GOTO 7
6: GOTO 23
```

```

7: IF i EQ 1 GOTO 9
8: GOTO 12
9: $t03 := total MULF 2.0
10: total := $t03
11: GOTO 14
12: $t04 := total SUBF 1.0
13: total := $t04

14: $t05 := CHSI i
15: i := $t05

16: IF total GTF 666.6 GOTO 20
17: GOTO 18
18: IF total EQ 500.0 GOTO 20
19: GOTO 5
20: $t06 := 1
21. i := $t06
22: GOTO 5

23: PARAM total
24: CALL PUTF,1
25: HALT

```

Lliurament

- Aquesta pràctica és individual.
- El lliurament es farà via Moodle, en les dates indicades al mateix.
- Cal lliurar un arxiu comprimit que contingui:
 - tot el codi font (sense arxius generats en el procés de compilació)
 - exemples i la seva sortida corresponent
 - scripts o Makefile per compilar-ho tot, executar el programa amb els exemples, i netejar-ho tot llevat dels arxius font i els d'exemple.
 - arxiu README (en format txt o pdf) amb instruccions per la compilació i execució, i descripció de tot allò que vulgueu destacar de la vostra pràctica (decisiones de disseny, funcionalitat addicional, limitacions, etc.).
- El nom de l'arxiu lliurat ha de contenir el vostre nom i primer cognom.