1 Pràctica III: Anàlisi semàntica i generació de codi intermedi

Alumnes: Joaquim Picó Mora, Sergi Simón Balcells
Professora: Maria Teresa Alsinet Bernadó
Curs: 2020-2021
GitHub

1.1 Característiques bàsiques

A continuació és mostra un llistat de les característiques principals que s'han implementat en aquest llenguatge:

- Tots els valors són constants
- S'ha implementat la definició de funcions (Vegeu a test/files/functions/normal_function.chs)
- S'ha implementat la crida de funcions (Vegeu a test/files/functions/call_func.chs)
- Addicionalment, s'ha implementat la currificació dels paràmetres de les funcions. (Vegeu a test/files/functions/currification.chs)
- No hi ha constants globals, només funcions globals. Tot s'ha de definir dins alguna funció.
- Àmbits de programa: S'ha pensat com una llista(pila) que resideix en la mònada AlexState en la qual es van afegint Maps a mesura que es creen contextos diferents.
- Suporta els tipus boleà, enter, caràcter i real.
- Es poden crear arrays i arrays n dimensionals (Vegeu a test/files/arrays)
- S'han creat totes les expressions d'enters, dels reals, dels boleans i algunes pel tipus caràcter. (Vegeu a test/files/expresions)
- S'ha creat l'estructura d'assignació per a les funcions i per a data.
- S'ha creat l'estructura if/else (Vegeu a test/files/conditional/conditional.chs)
- S'ha creat l'estructura while (Vegeu a test/files/while/while.chs)
- S'ha creat l'estructura for (Vegeu a test/files/for/for.chs)
- S'ha creat l'estructura repeat/until (Vegeu a test/files/repeat_until/repeat_until.chs)
- Addicionalment s'ha creat l'estructura map (Vegeu a test/files/map/map.chs)
- Addicionalment s'ha implementat la definició de tipus mitjançant data. (Vegeu a test/files/data/data.chs)
- Juntament amb data s'ha implementat l'estructura case of. (Vegeu a test/files/case_of/case_of.chs)

1.2 Three Address Code

Per tal de realitzar la generació de codi es crea un tipus ThreeAdressCode en el qual definim el llenguatge al qual traduïm. Tot el codi del nostre llenguatge es tradueix a aquest codi de tres adreces i posteriorment implementarem una instància de la Typclass Repr d'aquest tipus per tal de fer-ne la traducció a String senzilla. Aquesta instància de la Typeclass Repr farà la visualització d'aquest codi generat una tasca més còmode. Es pot veure la sintaxi del codi generat en els fitxers amb extensió
.tac de la carpeta test/files. En algun dels testos es proven casos d'error, i per tant en aquells no hi ha codi de tres adreces generat. S'ha de veure d'aquells en què no es prova un cas d'error.

1.2.1 Tests

Els tests s'han realitzat mitjançant una tècnica anomenada Golden Testing. Aquesta tècnica consisteix a facilitar els testos de programes que escriuen el seu resultat en fitxers. Aquesta tècnica de testing bolca els resultats dels programes en fitxers, per passar el test aquest resultat ha de ser idèntic que els resultats dels anomenats "Golden files", els quals contenen el resultat correcte del test.

1.3 Taula de símbols

En la mònada d'estat que ens dóna Àlex i emmagatzemem dos Maps. El map de definicions on s'hi guardaran les definicions de tipus que realitzem mitjançant data. I el Map de valors, on s'hi guardarà tota la resta de definicions.