

Progetto del Corso di Laboratorio di Linguaggi

Anno Accademico 2019/2020

Estensione della Semantica Dinamica di un Semplice Linguaggio Imperativo

Assegnamento

Si estenda l'interprete IMP presentato a lezione e scaricabile dall'elearning seguendo il percorso /Project/Base/imp.zip con i seguenti nuovi comandi:

- **Ciclo for.** Il ciclo `for` è una struttura di controllo iterativa la cui sintassi è data dalla seguente regola:

$$\text{for } (assign; exp; com_0) \{ com_1 \}$$

dove *assign* è un assegnamento arbitrario. La semantica dinamica di un generico ciclo `for` è equivalente al seguente comando:

$$assign; \text{while } (exp) \{ com_1; com_0 \}$$

Affinchè il costrutto `for` esegua senza errori, è necessario che *assign*, *com₀*, *com₁*, ed *exp* eseguano correttamente e che *exp* sia valutata ad un valore booleano.

- **Ciclo do-while.** Si implementi il costrutto iterativo `do-while` che esegue il suo corpo almeno una volta. La sua sintassi è

$$\text{do } \{ com \} \text{ while } (exp)$$

e la sua semantica risulta equivalente al seguente comando:

$$com; \text{while } (exp) \{ com \}$$

Il costrutto `do-while` esegue senza errori se *com* ed *exp* eseguono correttamente e se quest'ultima viene valutata ad un valore booleano.

- **Operatore condizionale if-then-elseif-else.** Si modifichi/estenda l'operatore condizionale `if-then-else` già presente nel linguaggio secondo la seguente regola sintattica:

$$\begin{aligned} &\text{if } (exp_0) \text{ then } \{ com_0 \} \\ &\text{elseif } (exp_1) \{ com_1 \} \\ &\dots \\ &\text{elseif } (exp_n) \{ com_n \} \\ &\text{else } \{ com_{n+1} \} \end{aligned}$$

I rami `elseif` ed `else` sono opzionali e possono essere omessi, mentre il ramo `if` è obbligatorio affinché il costrutto risulti ben formato. La semantica di un generico `if-then-elseif-else` è equivalente al

seguinte comando qualora sia presente almeno un ramo `elseif` o `else`:

```
if ( $exp_0$ ) then {  $com_0$  }
else {
    if ( $exp_1$ ) then {  $com_1$  }
    else {
        if ( $exp_2$ ) then {  $com_2$  }
        else {
            ...
        }
    }
}
```

Nel caso in cui sia presente solamente il ramo `if`, l'equivalenza è data rispetto al comando

```
if ( $exp_0$ ) then {  $com_0$  } else { skip }
```

L'operatore condizionale esegue correttamente quando tutte le espressioni exp_i ed i comandi com_i sono corretti. In particolare, le espressioni devono ritornare un booleano quando valutate.

■ **Scelta non deterministica nd.** L'operatore per la scelta non deterministica tra due comandi ha la seguente sintassi:

```
nd( $com_0$ ,  $com_1$ )
```

La sua semantica intuitiva è di eseguire *casualmente* uno tra i due comandi com_0 e com_1 . Affinchè esegua senza errori, devono essere corretti i due comandi utilizzati come argomenti.

Linee Guida

Dopo aver scaricato ed importato il progetto di partenza `imp.zip` in IntelliJ, lo svolgimento dell'elaborato può essere suddiviso in due fasi:

Estensione della grammatica: per ognuno dei costrutti da implementare, si estenda la grammatica del linguaggio in modo consistente con le regole sintattiche mostrate nella sezione precedente.

Implementazione della semantica dinamica: si esegua l'override dei nuovi metodi generati da ANTLR in seguito alla modifica della grammatica nella classe `IntImp` al fine di implementarne la semantica dinamica.

Testing

Una volta terminata l'implementazione, si eseguano dei test sia rispetto a programmi ben tipati che non per verificarne la correttezza. In particolare, una serie di programmi sono già stati caricati sull'elearning al percorso `/Project/Test`. I programmi all'interno della sottocartella `Well-Typed` sono programmi ben tipati che devono essere eseguiti dall'interprete senza nessun errore. Invece, i programmi all'interno della sottocartella `Bad-Typed` sono programmi mal tipati che devono stampare a video un errore durante la loro esecuzione.

Modalità di Consegna e Valutazione

Una volta terminato il progetto, si proceda alla consegna nel seguente modo: si esporti il progetto da IntelliJ (`File > Export to Zip file...`)¹ e lo si invii a `samuele.buro@univr.it` indicando il proprio

¹Se manca questa voce, create l'archivio `.zip` manualmente a partire dalla directory radice del progetto.

nome, cognome, e matricola. Se il progetto è stato svolto in coppia, inviate una singola email contenente anche i dati del vostro collega che metterete in cc.

Il progetto deve essere consegnato almeno una settimana prima del compito scritto che vorrete sostenere ed annulla un eventuale altro scritto svolto in precedenza. Se il progetto funziona correttamente (ovvero se esegue come previsto su tutti i test caricati sull'elearning), avrete la possibilità di rispondere ad una domanda riguardante l'implementazione del vostro progetto il giorno che vi presenterete a fare lo scritto. I punti che potrete ottenere sono

- 4, se il progetto è stato consegnato entro il 28/01 (indipendentemente da quando poi andrete a sostenere l'esame); oppure
- 3, negli altri casi.

Il progetto è consegnabile una sola volta (prima dello scritto, a meno che non decidiate di farvi annullare il voto) ed avrete una sola possibilità di rispondere alla domanda sul progetto il giorno che vi presenterete a fare lo scritto. Se la risposta non viene data o se prendete un punteggio che non vi soddisfa, non avrete la possibilità di ripresentarvi agli appelli successivi per rispondere nuovamente alla domanda sul progetto. Ovviamente, potrete ripetere lo scritto senza perdere il punteggio ottenuto sulla domanda riguardante il progetto.

I punti ottenuti dal progetto (max. 4) si sommano a quelli dello scritto (max. 27). Se la somma supera il 28 e lo studente vuole registrare un voto strettamente maggiore di 28, dovrà sostenere un esame orale, altrimenti il voto verrà troncato a 28. Una regola simile vale anche per chi non svolge il progetto: i punti ottenibili per chi ha deciso di non svolgere il progetto sono i soli dati dallo scritto (max. 27), e chi ha un voto superiore a 26 e vuole registrare un voto strettamente maggiore di 26, dovrà sostenere un esame orale.

Inoltre, per poter registrare il voto è necessario aver preso la sufficienza nel compito scritto.

Note su Come Importare il Progetto in IntelliJ

1. Scaricate il file `imp.zip` da `/Project/Base/imp.zip`.
2. Estraiete i file contenuti e prestate attenzione che non vengano create due cartelle annidate entrambe con il nome `imp`. Al termine dell'estrazione dovrete avere una cartella `imp` contenente solamente la cartella `src` e la cartella `lib`.
3. Dalla finestra iniziale di IntelliJ, scegliete `Import Project`. Se IntelliJ non vi apre la finestra iniziale ma l'ultimo progetto aperto, andate su `File > Close Project` e tornerete alla finestra iniziale.
4. Selezionate la cartella estratta `imp` e confermate.
5. Selezionate `Create project from existing sources` e proseguite fino al termine della procedura guidata, prestando attenzione a selezionare un SDK di Java >8.