

Tempo a disposizione: ore 2.

Svolgere gli esercizi 1–4, 5–6 e 7–8 su tre fogli separati.

Scrivere nome, cognome e matricola su ogni foglio consegnato.

- FOGLIO 1 ▷ 1. Considerando il semplice linguaggio di programmazione visto a lezione, definire le regole di semantica operativa strutturata per il costrutto $e_0 * e_1$ secondo la disciplina di valutazione interna-destra (ID). Quale risultato fornisce la valutazione ID dell'espressione $(3 - 5) * 0$?

- FOGLIO 1 ▷ 2. Dimostrare che $L = \{a^n b^n \mid n \geq 2\}$ è un linguaggio libero deterministico.

- FOGLIO 1 ▷ 3. Si consideri la grammatica G con simbolo iniziale S :

$$S \rightarrow aSc \mid aSb \mid \epsilon$$

(i) Quale linguaggio genera G ? (ii) Verificare se G sia di classe $LL(k)$ per qualche k .

- FOGLIO 1 ▷ 4. Costruire un parser SLR(1) per la grammatica G del punto precedente.

- FOGLIO 2 ▷ 5. Si consideri una funzione f con un parametro attuale X . Si dica, argomentando la risposta, se è possibile determinare staticamente il numero di valutazioni del parametro X fatte nel caso di una (singola) chiamata della funzione f nel caso in cui il parametro sia passato (i) per valore e (ii) per nome.

- FOGLIO 2 ▷ 6. Si considerino due linguaggi imperativi L1 ed L2 che non ammettono ricorsione. I comandi di L1 permettono assegnamento, comando condizionale ed iterazione determinata, quelli di L2 invece assegnamento, comando condizionale ed iterazione indeterminata. Si dica, motivando la risposta, se è possibile compilare (correttamente) da L2 a L1.

- FOGLIO 3 ▷ 7. Si consideri la funzione monomorfa $sum : Float \times Int \rightarrow Float$ che restituisce un numero a virgola mobile che è la somma tra i due numeri, uno a virgola mobile e l'altro intero, dati in input.

Scrivere versioni polimorfe di sum , considerando tre discipline diverse di polimorfismo e discuterne pro e contro. Per ciascuna versione, specificare se la versione polimorfa di sum può supportare l'utilizzo coi tipi in input $Int \times Float$ e $Int \times Int$ (eventualmente sfruttando il fatto che Int può essere convertito canonicamente a $Float$). Potrebbe essere usata con $String \times Int$? Come?

- FOGLIO 3 ▷ 8. Si consideri la seguente interfaccia per un contatore su numeri naturali, espressa come tipo esistenziale $Counter = \exists X. \{ start : Unit \rightarrow X, inc : X \times Nat \rightarrow X, get : X \rightarrow Nat \}$.

Fornire due implementazioni dell'interfaccia $Counter$. La prima, chiamata `SimpleCounter`, usa il tipo base Nat , che supporta l'operazione di somma $+$. L'altra, chiamata `LogCounter`, usa una lista di Nat (il cui tipo va definito) per tenere traccia dei singoli incrementi.

Per ciascuna implementazione, specificare il tipo concreto scelto per X e definire `start` (che fa partire da 0 il contatore), `inc` (che incrementa il contatore di tanto quanto specificato dal secondo parametro) e `get` (che ritorna il valore attuale del contatore), usando la sintassi `TipoImp, { op1(p1, ..., pi) = {...}, ..., opm(p1, ..., pj) = {...} } as TipoInterfaccia`.

Spiegare se è possibile far interagire le due implementazioni, ad esempio, dato `a` di tipo `SimpleCounter` e `b` di tipo `LogCounter`, è lecita l'istruzione `a.inc(b)`? Spiegare brevemente il ragionamento seguito.