

Relazione: secondo progetto intermedio

Programmazione II - Anno 2020/21

Il mio progetto è diviso in due file di codice sorgente:

- il primo, “Progetto PR2a typecheck dinamico”, contiene la realizzazione del linguaggio descritto nella consegna provvisto di funzione interprete, prevedendo typecheck dinamico.
- Il secondo, “Progetto PR2a typecheck statico”, contiene la realizzazione di un linguaggio che differisce dal primo solo per alcune espressioni, è però provvisto di una funzione che esegue il typecheck statico (richiesta nella consegna). Non sono previste però funzioni che svolgano compilazione o interpretazione.

Il motivo di questa scelta è che il primo linguaggio è strutturato per avere una funzione “eval” complessiva di interprete e controlli sintattici a tempo di compilazione. Prevede espressioni di creazione funzioni sprovviste dei tipi di valore in ingresso e valore in uscita. Questo non risulta essere un problema nel primo linguaggio, il compito del controllo dei tipi è lasciato alla valutazione di espressioni più interne rispetto a quelle di dichiarazione di funzioni (“Fun”, “Letrec”). Invece non è così per il secondo linguaggio che prevede un meccanismo di controllo dei tipi statico, privo di inferenza.

Il tipo set è implementato come una coppia (tipo di valori, lista di valori dello stesso tipo). La lista è sempre ordinata (secondo l’ordinamento degli operatori di confronto di Ocaml) per comodità di implementazione e per massimizzare l’efficienza di alcune funzioni come Unione, Intersezione ecc.

Ho aggiunto diverse funzioni di supporto per “eval”, per non creare un codice troppo sporco e per semplificarne la scrittura e la lettura. Ho aggiunto anche il tipo stringa per dare una varietà di tipi primitivi maggiore e alcune espressioni semplici (come Or logico o Decapitate) che sono utilizzate nella valutazione di espressioni più complesse come “ForAll” oppure “Filter”.

I test case sono compresi nei file sorgente, posti alla fine di essi. Per eseguire il progetto basta eseguire i due file separatamente e controllare che i test vadano a buon fine.