Progetto del nostro gruppo

Il nostro gruppo ha deciso di espandere il linguaggio che ci è stato fornito. Sono state effettuate le seguenti aggiunte:

* Integer, è stata aggiunta la possibilità di dichiarare variabili che contengono numeri interi
* Float, è stata aggiunta la possibilità di dichiarare variabili che contengono numeri floating point
* Boolean, è stata aggiunta la possibilità di dichiarare variabili che contengono valori booleani
* If
* Gli operatori logici (&&, ||, !)
* Simboli per la comparazione (==, !=, <=, >=)
* Radici e potenze.
* È stata implementata una symbol table tramite l’utilizzo di una linked list, i cui nodi possono salvare le seguenti informazioni: l’ID della variabile, il valore della variabile (tre tipi di valore che sono float, int e bool), il tipo della variabile e il puntatore al prossimo nodo
* Sono state create delle funzioni per accedere e modificare la symbol table:
  + funzione per aggiungere nuovi nodi
  + funzione per aggiornare il valore degli elementi esistenti
  + funzioni che ritornano il valore di un elemento richiesto
  + funzioni per stampare in output gli elementi della lista (o tutti o uno nello specifico)
  + funzioni che effettuano controlli di vario tipo:
    - controllo se la lista sia vuota
    - controllo su duplicati nella lista
    - controllo su possibili inconsistenze dei tipi delle variabili durante alcune operazioni (meccanismo di type checking)

Cosa fa il compilatore: il compilatore rappresenta una calcolatrice che può effettuare le classiche operazioni aritmetiche (addizione, sottrazione, moltiplicazione, divisione, potenza e radice) e anche operazioni logiche, come comparazione di valori (maggiore, maggiore uguale, minore, minore uguale, congruente e diverso) o operazioni di algebra booleana (and, or e not). Questa calcolatrice dà anche la possibilità di dichiarare variabili di tre tipi, ovvero intere, decimali o booleane. Il compilatore permette operazioni aritmetiche tra variabili intere e decimali come è permesso nei normali linguaggi di programmazione, mantenendo però il formato delle variabili sempre coerente con il loro tipo (quindi una variabile intera non avrà mai la virgola). Le operazioni tra tipi incompatibili vengono vietate grazie ad un meccanismo di type checking, che quindi evita operazioni aritmetiche tra variabili booleane, per esempio, oppure operazioni booleane tra variabili numeriche. Il linguaggio comprende anche un operatore condizionale if, che viene usato per decidere tra due valori sulla base di una condizione.

Oltre alle sopra elencate aggiunte è stata modificata la sintassi di base per poter effettuare il type checking.

L = {}

VT = {}

VNT = {}