Report Logo

Grammar:

Approccio:

22.05.2020

Installato ed impostato ant e java8

per prima cosa ho guardato come è il funzionamento del file Logo.jj.

Una volta capito questo, ho iniziato aggiungendo i due metodi per subroutine e statement.

Ho iniziato a tradurre la grammatica per subroutine e per gli statement e anche per tutti gli altri void come le expr.

24.05.2020

Il primo problema che ho avuto, è stato quando ho dovuto aggiungere il numero. Perché ero riuscito a scrivere tutte le espressioni ma ero bloccato alle espressioni Factor. Qui infatti non riuscivo a stampare il numero. Una volta riuscito a risolvere il problema, ho notato che per stampare il numero veniva chiamata la funzione due volte. Quindi, ho scritto una funzione in modo tale che dovessi richiamare solo questa per le volte che mi serviva e non scrivere tutto quanto tutte le volte.

Questo metodo l'ho usato anche per altre funzioni che nel codice si ripetevano, per non avere codice doppio inutile.

26.05.2020

Risolto questo problema, sono riuscito a far partire qualche logo file e sono riuscito a fare qualche test.

Un'altra difficoltà è stata per il Repeat infatti, sono riuscito a fare la funzione per un for normale, ma adesso sono bloccato per un nested for,sto cercando di capire come aggiungere una variabile in modo tale che quando viene richiamata la funzione la seconda volta la variabile non si sovrascrive.

02.06.2020

Ho creato un file test, per testare le operazioni di piu meno per diviso.

LOGO FACTOR  
 FD 1 + 2  
 FD 1 \* 1  
 FD 1 - 1  
 FD 1 / 1  
END

Dopo essere uscito a trovare una soluzione, ho testato le funzioni sul file poligono . Logo

ho avuto dei problemi nel file poligono. Logo, nel momento in cui non riuscivo a stampare il diviso.

private void poly(double SIDES){  
 for(int i = 0; i <= SIDES; i++){  
 logo.fd(50);  
 logo.rt(360SIDES);  
 }  
}

dopo

logo.rt(360/SIDES);

Funziona tutto.

09.06.2020

Ho avuto problemi con per la parte REPEAT:

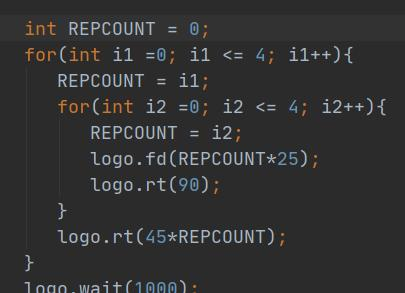
<REPEAT>{indent(); pw.print("for(int i = 0; i <= ");} nexpr() {pw.println("; i++){");numIndent++;} <LBRA> (statement())\* <RBRA> {numIndent--; indent();pw.println("}");}

Questa è stata la mia prima prova per capire se funzionava, e per i cicli for singoli funziona benissimo

Il problema si pone per i cicli nested.

Ho dichiarato una variabile REPCOUNT:

Ho applicato un counter in modo da aggiungere ad i un numero:



Il mio problema in questo momento è che non riesco a decrementare REPCOUNT correttamente, mentre il contatore nessun problema:

