Για να υλοποιήσουμε τα ζητούμενα χρησιμοποιήσαμε το flex και bison πρόγραμμα που υλοποιεί την διαδικασία για να έχουμε ένα ελεγκτή σου συντακτικού μιας γλώσσας που μας δόθηκε.

H γλώσσα μας θα καλύπτει τα παρακάτω:

1. To βασικό κομμάτι του προγραμματος
2. Δήλωση μεταβλητών
3. Δήλωση συναρτήσεων
4. Structs
5. Όλες τις εντολές που αναφέρει η εκφώνηση
6. Σχόλια

Για την δημιουργία της παραπάνω διαδικασίας δημιουργήσαμε πρώτα τον συντακτικό αναλυτή που εισάγεται στο bison όπου ουσιαστικά αποτελεί και το bnf κώδικα στο μέρος της συντακτικής ανάλυσης (αρχείο bison.y)

Επίσης δημιουργήθηκε και ο λεκτικός αναλυτής που δίνεται στο αρχείο flex.l

Η συντακτική ανάλυση είναι η παρακάτω

|  |
| --- |
| **start\_rule: c\_\_PROGRAM\_\_ METAVLITI diloseis\_arxikes vasiko ;**  **diloseis\_arxikes: diloseis**  **| diloseis\_arxikes diloseis;**  **diloseis: synartiseis**  **| domes |;**  **synartiseis: synartisi**  **| synartiseis synartisi;**  **synartisi: c\_\_FUNCTION\_\_ METAVLITI '(' metavlites ')'**  **diloseis\_metavlitwn**  **seira\_entolwn**  **c\_\_RETURN\_\_ ekfrasi\_arithmitiki ';'**  **c\_\_END\_FUNCTION\_\_ ;**  **domes: entoli\_struct**  **| domes entoli\_struct;**    **vasiko: c\_\_STARTMAIN\_\_ diloseis\_metavlitwn seira\_entolwn c\_\_ENDMAIN\_\_ ;**    **entoli\_struct: c\_\_STRUCT\_\_ METAVLITI diloseis\_metavlitwn c\_\_ENDSTRUCT\_\_ ';' |**  **c\_\_TYPEDEF\_\_ c\_\_STRUCT\_\_ METAVLITI diloseis\_metavlitwn c\_\_ENDSTRUCT\_\_ ';' ;**  **diloseis\_metavlitwn: | seira\_vars**  **| diloseis\_metavlitwn seira\_vars ;**      **seira\_vars: c\_\_VARS\_\_ typos**  **metavlites\_me\_kommata ';';**  **typos: c\_\_INTEGER\_\_ | c\_\_CHAR\_\_;**  **metavlites: | metavlites\_me\_kommata;**  **metavlites\_me\_kommata: metavl\_2**  **| metavlites\_me\_kommata ',' metavl\_2;**  **metavl\_2: METAVLITI**  **| METAVLITI '[' c\_\_NUMBER\_\_ ']';**  **seira\_entolwn: mia\_entoli |**  **seira\_entolwn mia\_entoli;**  **mia\_entoli: anathesi\_timis|**  **while | print |**  **for | if\_entoli |**  **switch\_entoli|**  **c\_\_BREAK\_\_ ;**  **anathesi\_timis: METAVLITI '=' ekfrasi\_arithmitiki ';' ;**  **while: c\_\_WHILE\_\_ '(' syntikes ')' seira\_entolwn c\_\_ENDWHILE\_\_;**  **print: c\_\_PRINT\_\_ '(' c\_\_CHR\_\_ ')'';' |**  **c\_\_PRINT\_\_ '(' c\_\_CHR\_\_ ',' ekfrasi\_arithmitiki ')'';';**  **for: c\_\_FOR\_\_ METAVLITI ':' '=' ekfrasi\_arithmitiki c\_\_TO\_\_ ekfrasi\_arithmitiki c\_\_STEP\_\_ ekfrasi\_arithmitiki**  **seira\_entolwn**  **c\_\_ENDFOR\_\_;**  **if\_entoli: c\_\_IF\_\_ '(' syntikes ')' c\_\_THEN\_\_**  **seira\_entolwn**  **else\_elseif**  **c\_\_ENDIF\_\_ ;**        **else\_elseif: elseif\_kommati else\_kommati**  **| else\_kommati |;**        **else\_kommati: c\_\_ELSE\_\_ seira\_entolwn ;**  **elseif\_kommati: elseif1 | elseif\_kommati elseif1;**  **elseif1: c\_\_ELSEIF\_\_ '(' syntikes ')';**  **switch\_entoli: c\_\_SWITCH\_\_ '(' ekfrasi\_arithmitiki ')'**  **case\_kommati**  **default\_entoli**  **c\_\_ENDSWITCH\_\_ ;**  **case\_kommati: case**  **| case\_kommati case ;**    **case: c\_\_CASE\_\_ '(' ekfrasi\_arithmitiki ')'**  **seira\_entolwn ;**      **default\_entoli: | c\_\_DEFAULT\_\_ seira\_entolwn;**  **syntikes: ekfrasi\_arithmitiki symvola\_sygkrisewn ekfrasi\_arithmitiki |**  **syntikes and\_or syntikes ;**  **symvola\_sygkrisewn: '>'**  **| '<' |**  **c\_\_GREATER\_\_**  **| c\_\_LOWER\_\_**  **| c\_\_EQUAL\_\_**  **|c\_\_NOTEQUAL\_\_;**    **and\_or: c\_\_AND\_\_**  **| c\_\_OR\_\_;**  **ekfrasi\_arithmitiki: praxeis | ekfrasi\_arithmitiki praxi ekfrasi\_arithmitiki ;**  **praxeis: ekfrasi\_metv |**  **'(' ekfrasi\_arithmitiki ')' ;**  **praxi: '+'|**  **'-'|**  **'/'|**  **'\*'|**  **'^';**  **ekfrasi\_metv: METAVLITI**  **| METAVLITI '(' ekfrasi\_metv ')'**  **| METAVLITI '[' ekfrasi\_arithmitiki ']'**  **| c\_\_NUMBER\_\_**  **| c\_\_CHR\_\_ ;** |

Παρατίθεται μαζί με το πρόγραμμα τα αρχεία του bison και flex τα οποία δοκιμάστηκαν στο παρακάτω πρόγραμμα:

|  |
| --- |
| PROGRAM prg1  STARTMAIN  VARS INTEGER a;  a=10;  PRINT ("hello",a);    FOR i:=2 TO 4 STEP 1  a=a+1;  ENDFOR  a=a+10;    IF(a>3) THEN  a=a+1;  ENDIF        ENDMAIN |

Τελικά το αποτέλεσμα ήταν το παρακάτω:

