

Tema 2 – Construirea unui compilator pentru un mini-limbaj de programare

Enunț

Să se creeze un compilator pentru un mini-limbaj de programare, care să analizeze și să valideze un program scris în mini-limbajul respectiv. Programul include declarații de funcții și de variabile, comentarii, structuri decizionale, structuri repetitive și expresii (aritmetice, relaționale, logice, etc.).

Cerințe

I. Programul sursă va fi citit dintr-un fișier text.

II. Mini-limbajul va include următoarele categorii de unități lexicale:

- identificatori: denumiri pentru variabile și funcții;
- constante numerice și literali (șiruri de caractere cuprinse între ghilimele);
- cuvinte cheie: `int`, `float`, `double`, `string`, `void`, `if`, `else`, `for`, `while`, `return`;
- operatori aritmetici: `+`, `-`, `*`, `/`, `%`;
- operatori relaționali: `<`, `>`, `<=`, `>=`, `==`, `!=`;
- operatori logici: `&&`, `||`, `!`;
- operatori de atribuire: `=`, `+=`, `-=`, `*=`, `/=`, `%=`;
- operatori de incrementare `++` și decrementare `--`;
- delimitatori: paranteze (`(`, `)`, `{`, `}`), virgulă (`,`), punct și virgulă (`;`).

III. Compilatorul va procesa programul sursă pentru a realiza următoarele operațiuni:

1. Neglijarea spațiilor albe și a comentariilor de tip linie (`//`) și de tip bloc (`/* */`).
2. Obținerea și salvarea într-un fișier text a listei de unități lexicale. Fiecare element al listei este de forma `<token, lexemă, linie>`.

3. Colectarea și salvarea în fișiere text distincte a elementelor de sintaxă prezente în program.
 - (a) O listă de variabile globale (împreună cu tipul acestora + valorile cu care au fost inițializate la momentul declarării);
 - (b) O listă de funcții. Fiecare element al listei va conține:
 - tipul funcției (iterativă sau recursivă, `main` sau nu), tipul returnat, numele și lista parametrilor;
 - lista variabilelor locale (împreună cu tipul acestora + valorile cu care au fost inițializate la momentul declarării);
 - structurile de control prezente în funcție (`if-else`, `for`, `while`).
4. Semnalarea eventualelor erori lexice, sintactice și semantice.

Puteți scrie clase și funcții auxiliare ce sunt necesare pentru rezolvarea acestor cerințe.

Barem

1. Definirea unităților lexice ale mini-limbajului - 1.5p
2. Neglijarea spațiilor albe și a comentariilor - 0.5p
3. Obținerea listei de unități lexice - 0.5p
4. Definirea regulilor sintactice ale mini-limbajului - 2.5p
5. Generarea listei de variabile globale - 0.5p
6. Generarea listei de funcții și a detaliilor asociate - 2p
7. Semnalarea erorilor - 1.5p

Cerință pentru bonus de 0.25p la nota finală de laborator

Ca funcționalitate suplimentară, compilatorul va analiza și valida declarații de structuri (`struct`). Fiecare structură poate conține câmpuri, funcții membre, constructori și destructor.