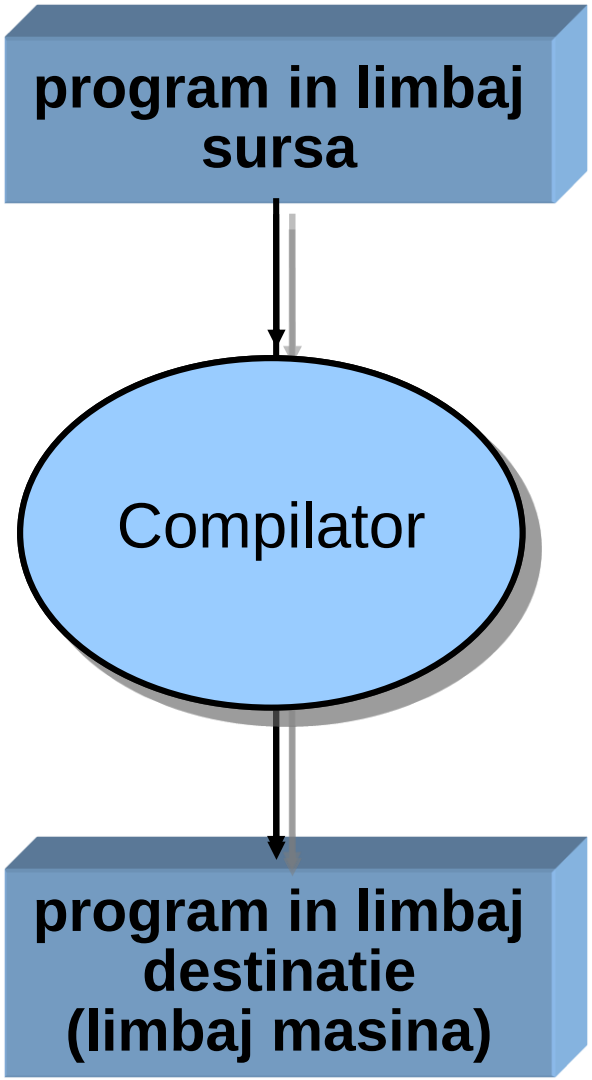
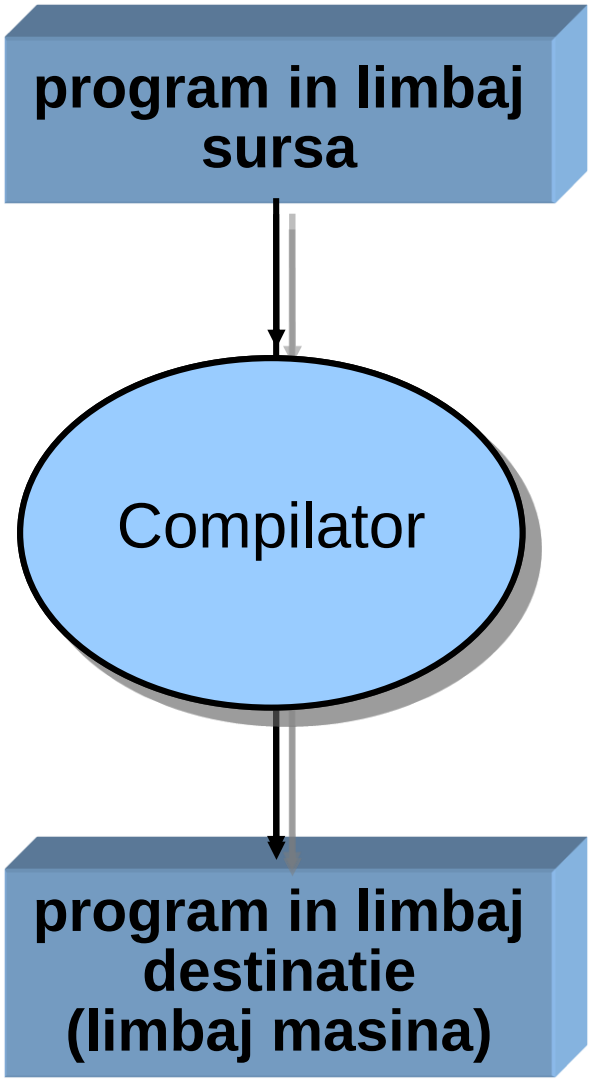
Compilator



- **Compilator**:
- **translateaza** informatiile dintr-un limbaj sursa intr-un limbaj destinatie echivalent
  - Limbajul sursa este superior limbajului destinatie
- **determina** cea mai buna translatare

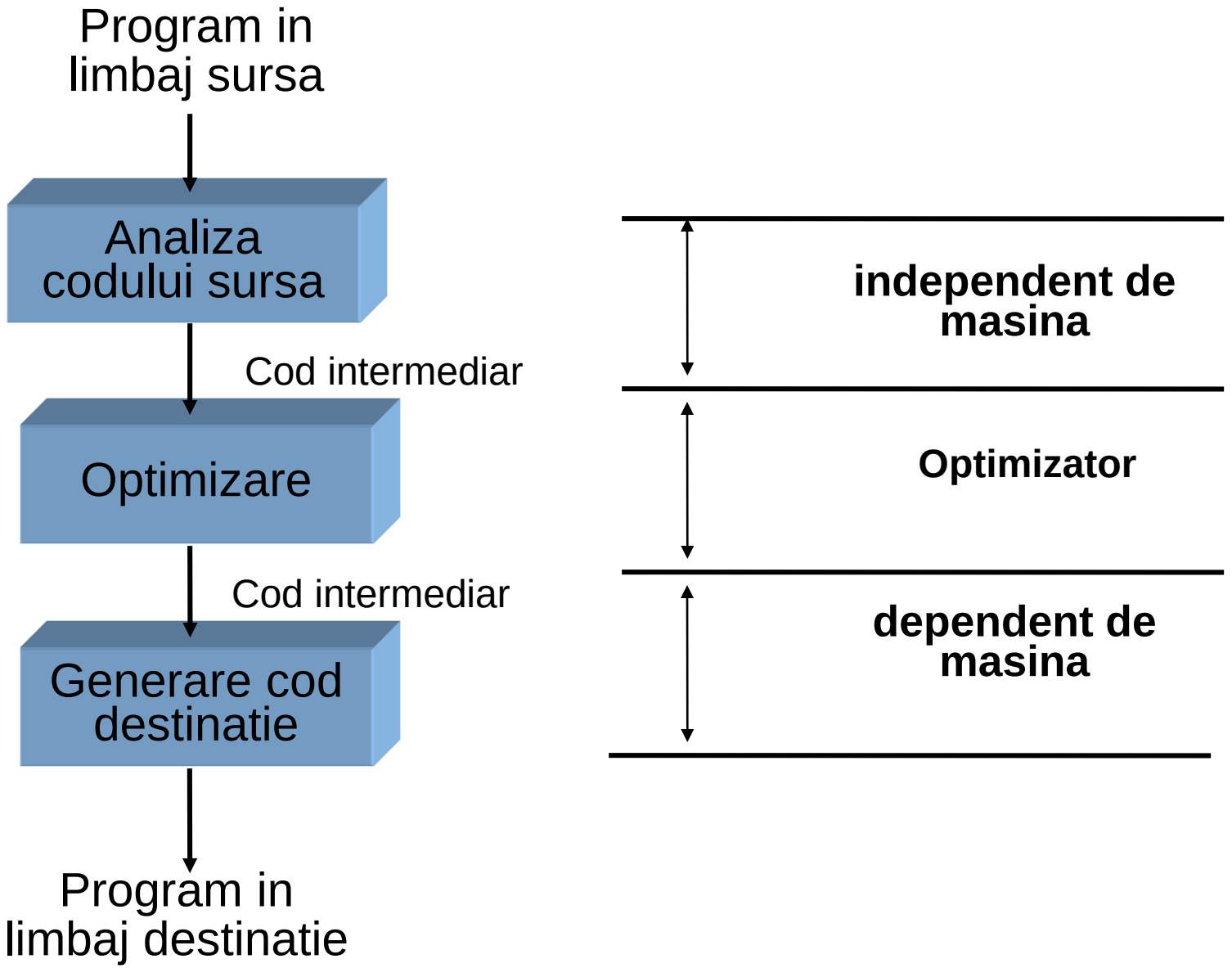
Compilator




```
float a, b, c;
a = b + c * 60;
```

```
MOVF Id3, R2
MULF 60.0, R2
MOVF Id2, R1
ADDF R2, R1
MOVF R1, Id1
```

Compilator


Program in limbaj sursa
Analiza codului sursa
Cod intermediar
Optimizare
Cod intermediar
Generare cod destinatie
Program in limbaj destinatie
independent de masina
Optimizator
dependent de masina

Compilator

```
float a, b, c;
a = b + c * 60;
```

Identifica cuvinte, numere, operatori si separatori

Analiza lexicala

float	a
a	=
,	b
b	+
,	c
c	*
;	60
	;

Compilator

```
float a, b, c;
a = b + c * 60;
```

Analiza lexicala

Tip ID V ID V ID PV

ID ATRIB ID PLUS ID INM NR PV

Etapele unui compilator

float a, b, c;
a = b + c * 60; ⟶ Tip ID V ID V ID PV

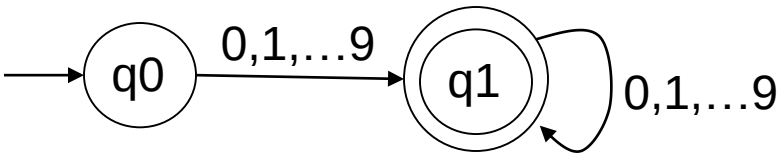
ID ATRIB ID PLUS ID INM NR PV

Analiza lexicala – identificare atomi lexicali: expresii regulate, automate finite

Numar:

Expresie regulata: (0+1+2+…+9)(0+1+2+…9)*

Automat finit determinist



float a, b, c;
a = b + c * 60;

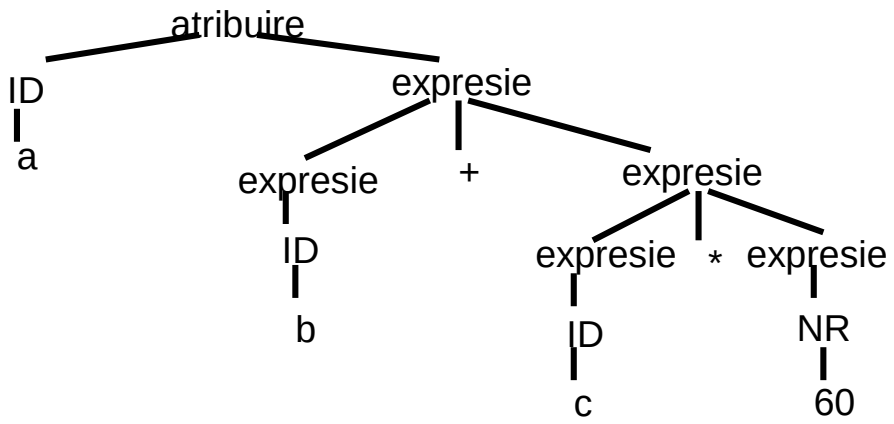
Compilator

Tip ID V ID V ID PV

ID ATRIB ID PLUS ID INM NR PV

Analiza sintactica

instructiune: ID atribuire expresie
expresie: expresie + expresie
expresie * expresie
ID
NR



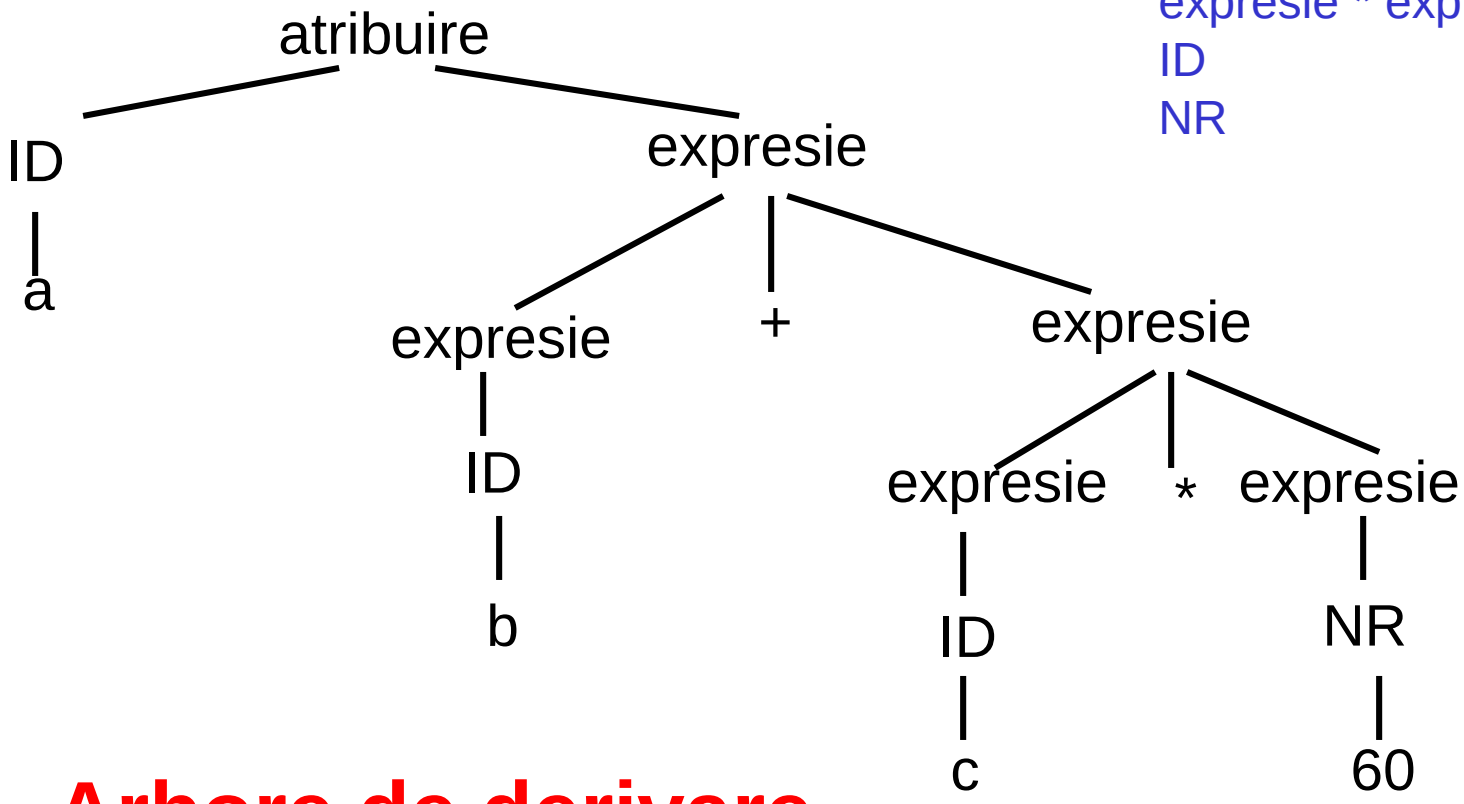
Etapele unui compilator

Analiza sintactica

a = b + c * 60 → ID ATRIB ID PLUS ID INM NR

Gramatica Independenta de context:

instructiune: ID atribuire expresie
expresie: expresie + expresie
expresie * expresie
ID
NR

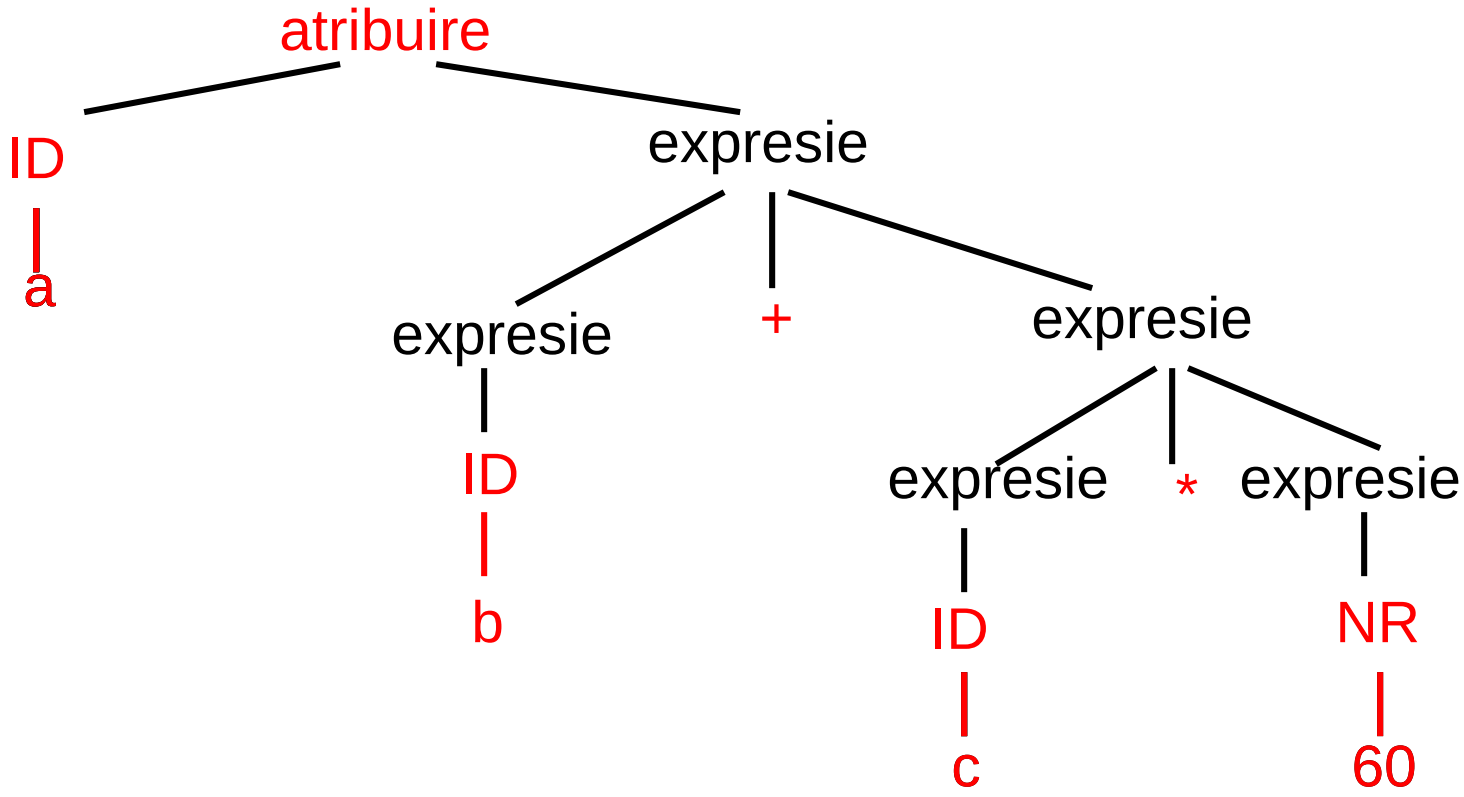


Arbore de derivare

Etapele unui compilator

Analiza sintactica

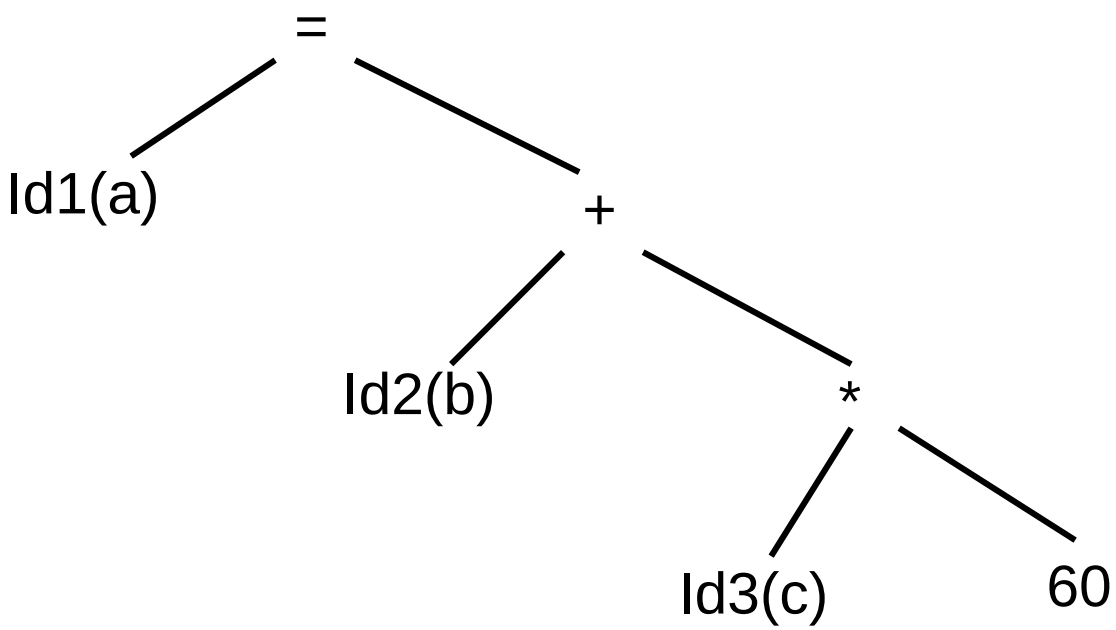
Arbore de derivare



Etapele unui compilator

Analiza sintactica

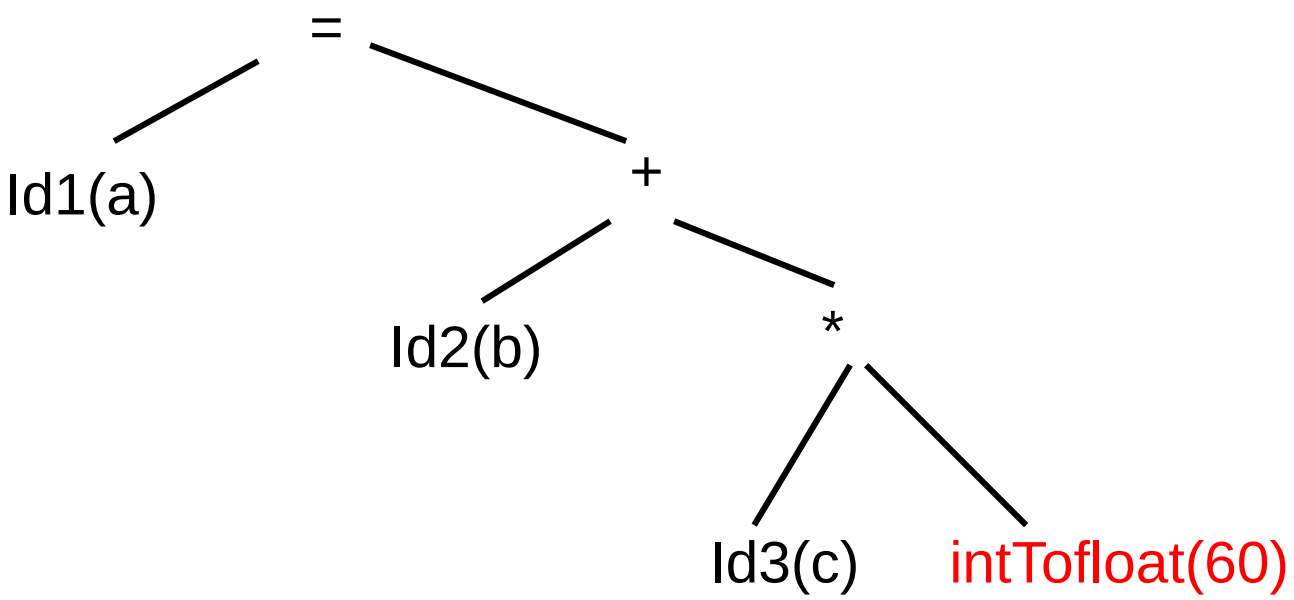
Arbore sintactic



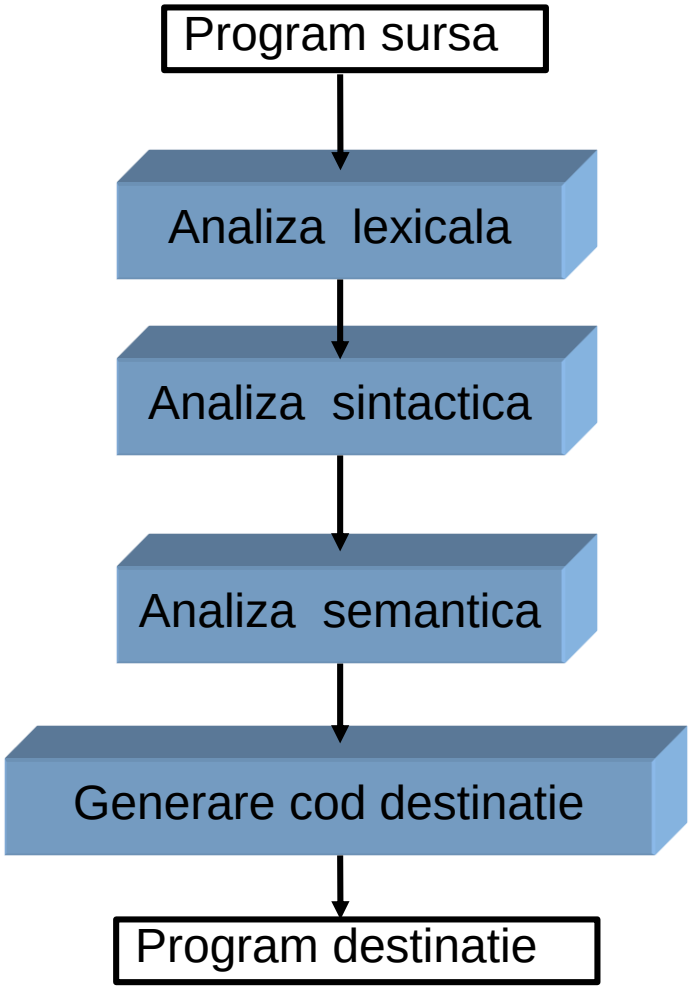
Etapele unui compilator

Analiza semantica

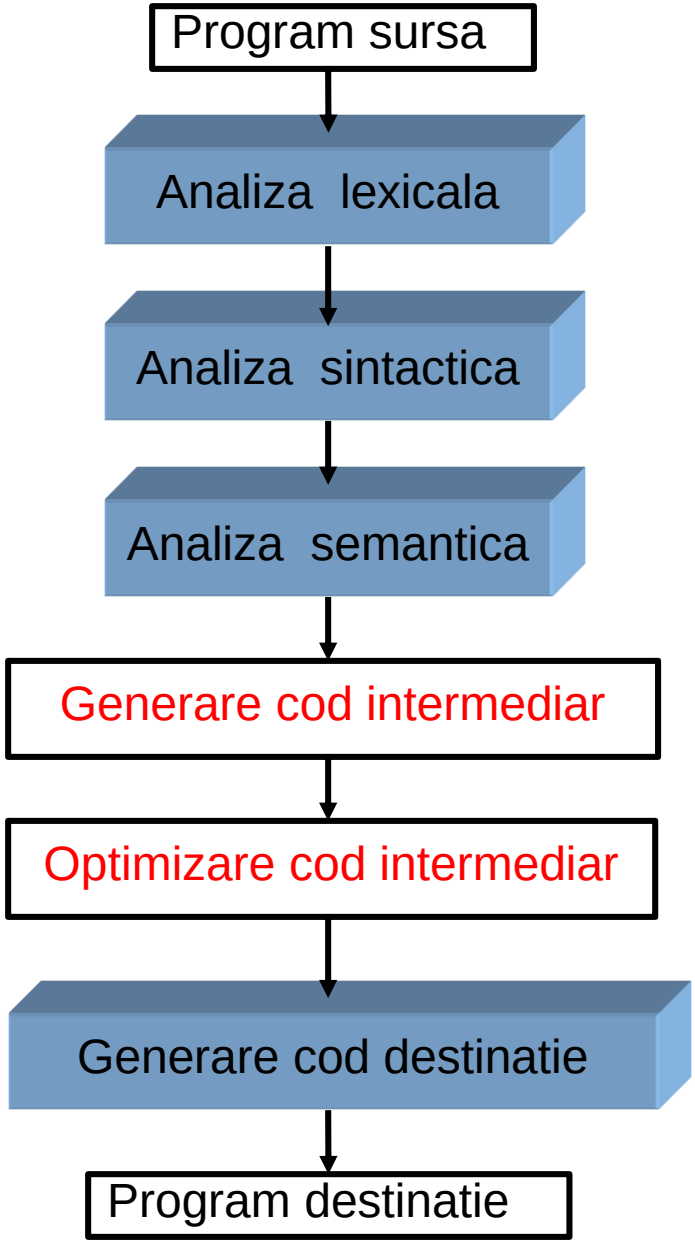
Arbore sintactic - verificare de tipuri



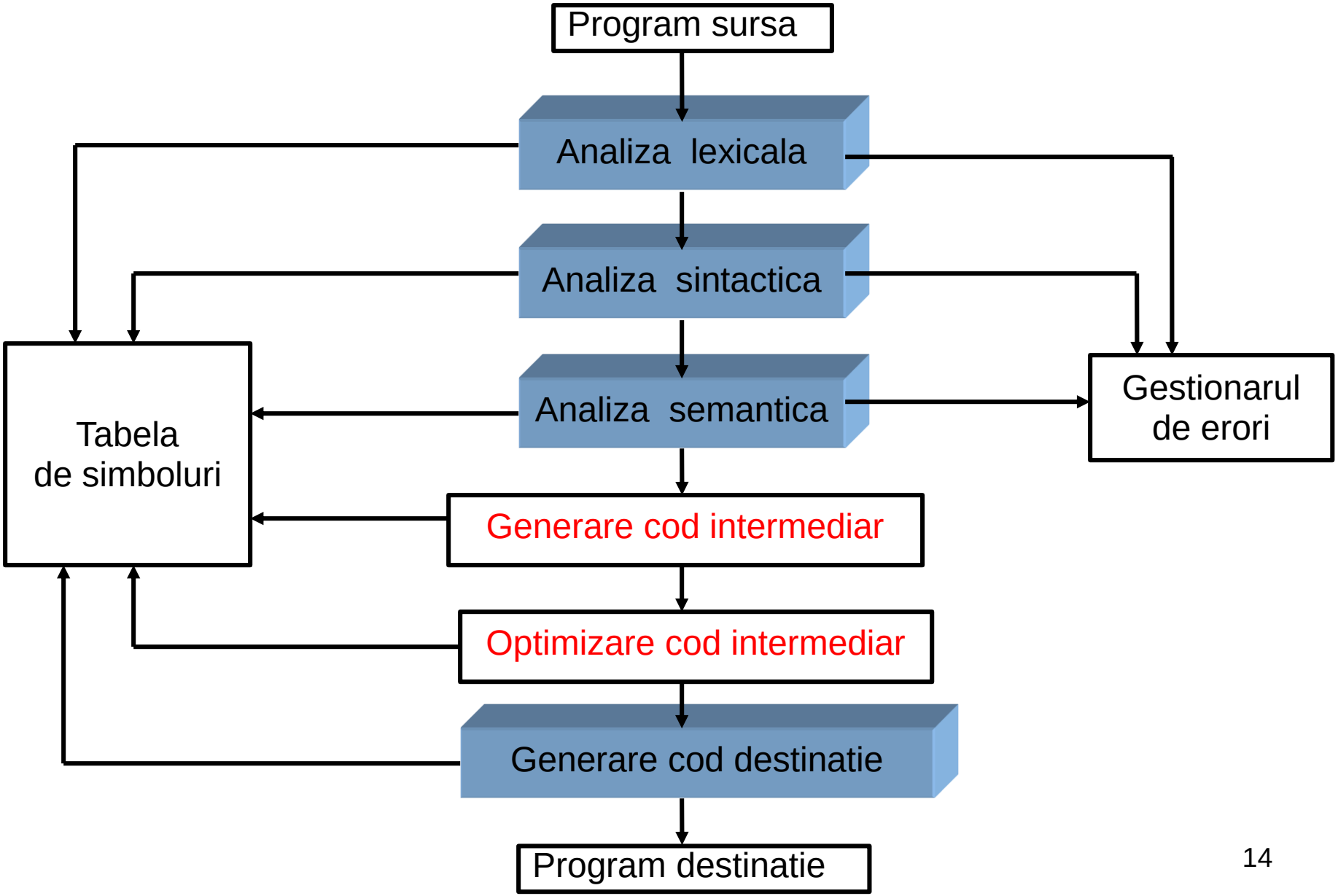
Strucura generala a unui compilator



Strucura generala a unui compilator

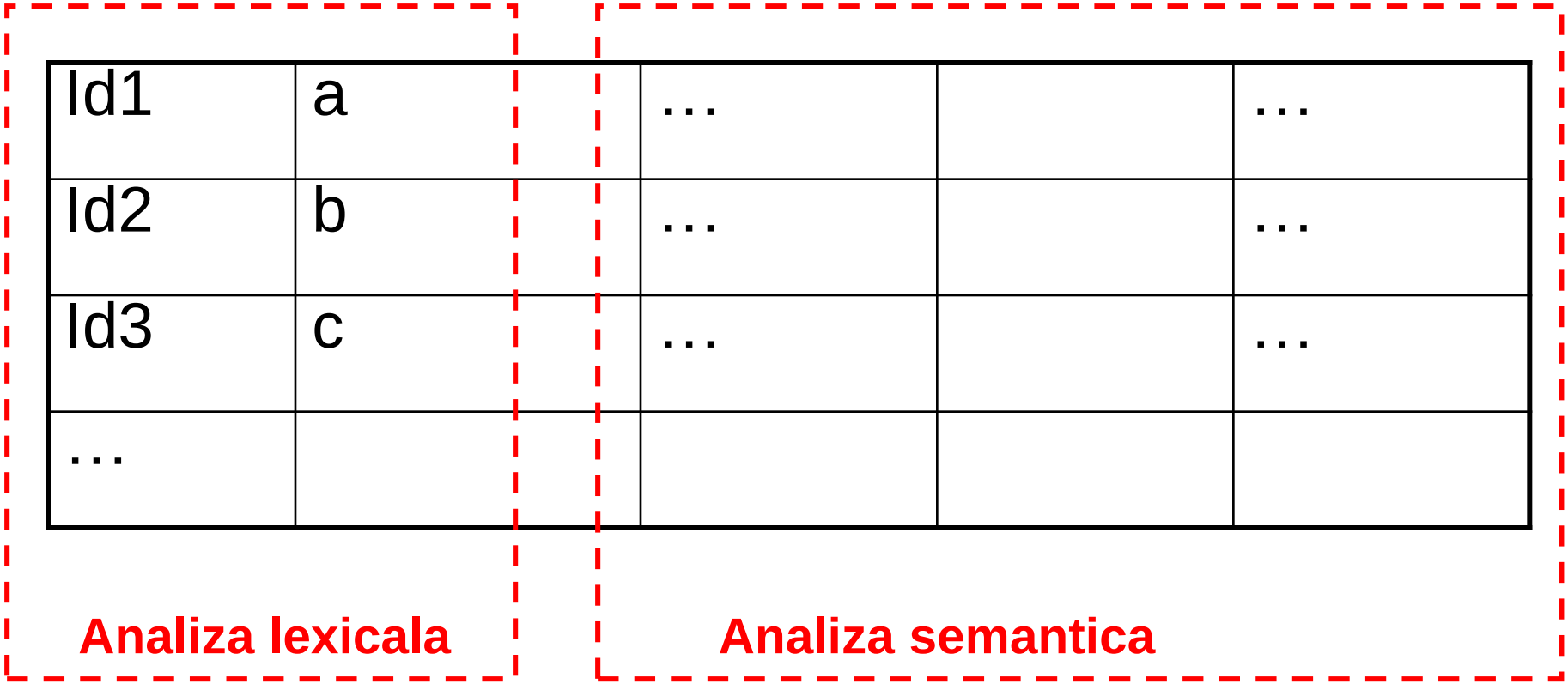


Strucura generala a unui compilator



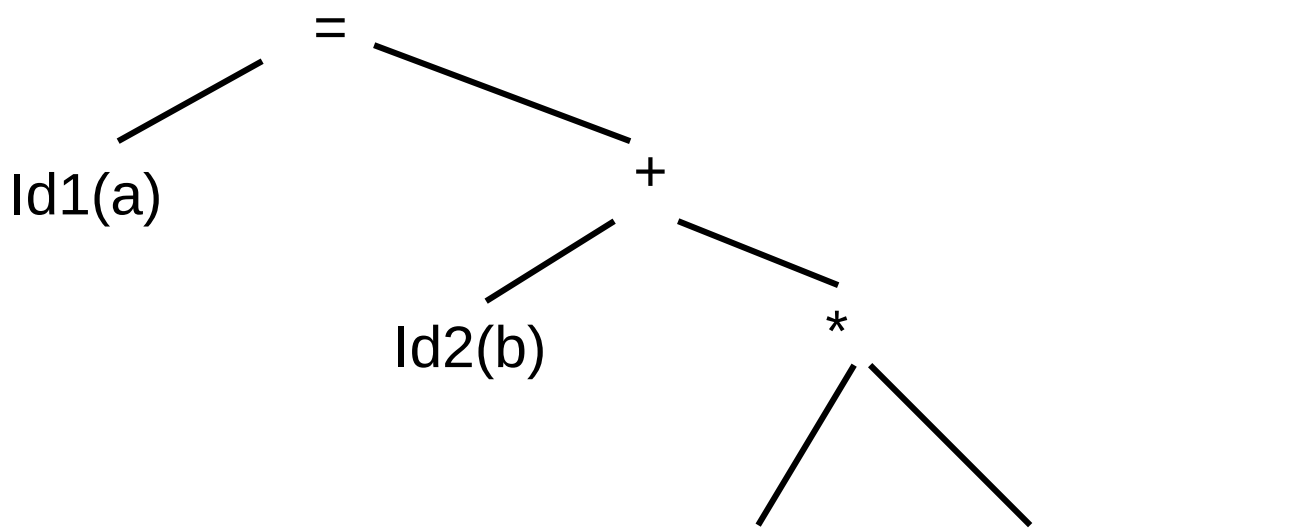
Etapele unui compilator

Gestiunea tabelei de simboluri



Etapele unui compilator

Generare de cod intermediar



temp1 = intTofloat(60)

temp2 = id3*temp1

temp3 = id2+temp2

Id1 = temp3

Etapele unui compilator

Optimizarea codului intermediar

temp1 = intTofloat(60)
temp2 = id3*temp1
temp3 = id2+temp2
Id1 = temp3

temp1 = intTofloat(60)
temp2 = id3*temp1
temp3 = id2+temp2
Id1 = temp3

temp1 = id3 * intTofloat(60)
Id1 = id2 + temp1

Etapele unui compilator

Generarea de cod

temp1 = id3 * intTofloat(60)

Id1 = id2 + temp1

Instructiuni de tip:

Instr sursa, destinatie

Registri: R1, R2

MOVF Id3, R2

MULF 60.0, R2

MOVF Id2, R1

ADDF R2, R1

MOVF R1, Id1