**1. Specificarea minilimbajului de programare (MLP)**

<program>::=<decl\_functii>

<decl\_functii>::=<functie><decl\_functii>|<functie>

<functie>::=<antet><corp>

<antet>::=<tip> ID (<lista\_decl>)

<tip>::=int | float | double | <struct>

<lista\_decl>::=<decl>,<lista\_decl>|<decl>

<decl>::=<tip> ID

<corp>::={<instr\_comp>}

<instr\_comp>::=<instr><instr\_comp>|<instr>

<instr>::=<atr>;|<decl>;|<instr\_if>|<instr\_cicl>|<instr\_intrare>|<instr\_iesire>|<instr\_return>|<apel\_func>

<atr>::=ID = <expr\_arit>

<expr\_arit>::=<expr\_arit><op\_arit><expr\_arit>| ID | CONST | <apel\_func>;

<op\_arit>::= +|-|\*|/

<instr\_if>::= if (<cond>) <corp> | if (<cond>) <corp> else <corp>

<cond>::=ID|CONST|<expr\_arit><op\_rel><expr\_arit>|<expr\_arit>

<op\_rel>::= != | == | <= | >= | < | >

<instr\_cicl>::= while (<cond>) <corp>|for (<atr>;<cond>;<atr>) <corp>

<instr\_intrare>::= cin <lista\_intrari>;

<lista\_intrari>::= >> ID <lista\_intrari> | >> ID

<instr\_iesire>::= cout <lista\_iesiri>;

<lista\_iesiri>::= << <expr\_arit> <lista\_iesiri> | << <expr\_arit>

<instr\_return>::= return <expr\_arit>;

<apel\_func> ::= ID (<lista\_expresii>)

<struct>::= struct { <lista\_decl\_struct> }

<lista\_decl\_struct>::= <decl>;<lista\_decl\_struct>|<decl>;

<lista\_expresii>::= <expr\_arit>,<lista\_expresii>|<expr\_arit>

**2a. calculeaza perimetrul si aria cercului de o raza data**

int main() {

float pi ;

pi = 3.14159265359 ;

float raza ;

cin >> raza ;

float perimetru ;

perimetru = 2 \* pi \* raza ;

float arie ;

arie = pi \* raza \* raza ;

cout << perimetru ;

cout << arie ;

return 0 ;

}

**2b. determina cmmdc a 2 nr naturale**

int main() {

int a ;

int b ;

cin >> a ;

cin >> b ;

while ( a != b ) {

if ( a > b )

a = a – b ;

else

b = b – a ;

}

cout << a ;

return 0 ;

}

**2c. calculeaza suma a n numere citite de la tastatura**

int main() {

int n ;

cin >> n;

int suma ;

suma = 0 ;

for ( int i = 0 ; i < n ; I = I + 1 ) {

int num ;

cin >> num ;

suma = suma + num ;

}

cout << suma ;

return 0 ;

}

**3a. doua erori care sunt in acelasi timp erori in limbajul original**

int main() {

int x = 10 **// lipseste ;**

**int** y = **"abc"** ;

return 0 ;

}

**3b. doua erori conform MLP, dar care nu sunt erori in limbajul original**

**#include <iostream> // nu pot avea include in MLP**

int main() {

int x = 10 ;

int y = 5 ;

cout << x **%** y ; **// nu pot avea % in MLP**

return 0 ;

}