

<https://github.com/lavrric/UniversityProjects/tree/main/FLCD/lab8>

```
@echo off
lex spec.lxi
gcc lex.yy.c -o lexus -- makes an executable
if not ERRORLEVEL 1 (
    .\lexus.exe ../test_programs/p1.txt -- executes it
) else (
    -- in case of errors on compiling also prints the errors
    echo !!!! !!!! Lex compiling failed !!!! !!!!
)
```

ON p1.txt:

```
x = Number(in("Enter x: "));
y = Number(in("Enter y: "));
z = Number(in("Enter z: "));
```

```
if(x >= y && x >= z){
    out(x);
};
elif(y >= x && y >= z){
    out(y);
};
elif(z >= x && z >= y){
    out(z);
};
```

PRINTS:

Id: x

Operator: =

Reserved word: Number

Separator: (

Reserved word: in

Separator: (

String Constant: "Enter x: "

Separator:)

Separator:)

Separator:)

Separator: ;

Id: y

Operator: =

Reserved word: Number

Separator: (

Reserved word: in
Separator: (
String Constant: "Enter y: "
Separator:)
Separator:)
Separator:)
Separator: ;
Id: z
Operator: =
Reserved word: Number
Separator: (
Reserved word: in
Separator: (
String Constant: "Enter z: "
Separator:)
Separator:)
Separator:)
Separator: ;
Reserved word: if
Separator: (
Id: x
Operator: >=
Id: y
Operator: &&
Id: x
Operator: >=
Id: z
Separator:)
Separator: {
Reserved word: out
Separator: (
Id: x
Separator:)
Separator: ;
Separator: }
Separator: ;
Reserved word: elif
Separator: (
Id: y
Operator: >=
Id: x
Operator: &&
Id: y
Operator: >=

Id: z
Separator:)
Separator: {
Reserved word: out
Separator: (
Id: y
Separator:)
Separator: ;
Separator: }
Separator: ;
Reserved word: elif
Separator: (
Id: z
Operator: >=
Id: x
Operator: &&
Id: z
Operator: >=
Id: y
Separator:)
Separator: {
Reserved word: out
Separator: (
Id: z
Separator:)
Separator: ;
Separator: }
Separator: ;