STUDENT NUMBER [_____] NAME [_____]

ONE. Consider the following expression language based on the basic language we have studied in the course, but extended with the following expression forms

E :=	**....**	% other constructs
|	E>E	% greather than
|	E==E	% equality of integer values
|	**not** E	% negation
|	E **and** E	% conjunction
|	E **? (** E **:** E**)**	% conditional expression
|	**for** x=E **to** E **sum** E **end**	% conditional expression

In this language, like in the C language, "boolean" values are represented by integer values, where 0 represents "false" and any non-zero value represents "true".

The expression form E1 **? (** E2 **:** E3**)** evaluates E1 and if the result represents true, it returns the value of E2, otherwise it returns the value of E3.

In the expression form **for** x=E1 **to** E2 **sum** E3 **end** the identifier x is local with scope E3.

The value of **for** x=E1 **to** E2 **sum** E3 **end** is obtained by first evaluating E1 and E2, obtaining integers v1 and v2, and then returning the sum of expression E3 considering the value of x in E3 to vary between v1 and v2.

So for example, we expect the following evaluations

5 > 2 evaluates to 1

let x = 5 **in** x > 5 **end** evaluates to 0

2 > 9 ? (2+4: 2-4) evaluates to -2

for x = 1 **to** 4 **sum** x **end** evaluates to 10 (e.g. 1+2+3+4).

let y = 0
  in
  for x = y+1 **to** 4 **sum x+y end**
end evaluates to 10

1. Explain the diference between concrete syntax and abstract syntax.

2. Define the five Java classes necessary to represent the abstract syntax for the five constructs above.

All classes should implement the interface **ASTNode**, and support an evaluation method (interpreter) with the following signature.

int eval(Environment e);

TWO. Consider the job of designing an interpreter and compiler for the language we have studied till now in the course.

Consider the following program.

let x = 5 y = 0 **in**
 let x = x+y **in**
   2*x+y
 end
end

0. Indicate the result of evaluating the program above.

1. Explain (draw) what is the state of the **interpreter** environment at the time expression 2*x+y is evaluated

2. Explain carefully what is the JVM code generated for the expression 2*3+4.

3. Explain carefully what is the JVM code generated for the program above. Don't forget to include the generated code for the necessary stack frames.

NOTES:

You may consult whatever material you want.

Don't forget to write your name and number in the test sheet.

Don't forget to number your answers in the following sheets.