


CHALMERS

EXAMINATION / TENTAMEN

Course code/kurskod		Course name/kursnamn		
DAT151		Programming Language Technology		
Anonymous code Anonym kod		Examination date Tentamensdatum	Number of pages Antal blad	Grade Betyg
DAT151-3		2017-01-11	5	4

Solved task Behandlade uppgifter	Points per task Poäng på uppgiften	Observe: Areas with bold contour are to completed by the teacher. Anmärkning: Rutor inom bred kontur ifylles av lärare.
No/nr		
1	X	7
2	X	9
3	X	7.5
4	X	6
5	X	11.5
6		
7		
8		
9		
10		
11		
12		
13		
14		
15		
16		
17		
18		
Total examination points Summa poäng på tentamen	41	

Family name+First name (Blockletters) Efternamn+Förnamn+Initialer(textas)	THORSELL ERIK E.T	
Signature Namnteckning		
Year of Admission Antagningsår	2017	
Programme acronym Program	-	
Identification no nummer	-	
Date of Birth Year Month Day Personnummer år mån dag	- - -	

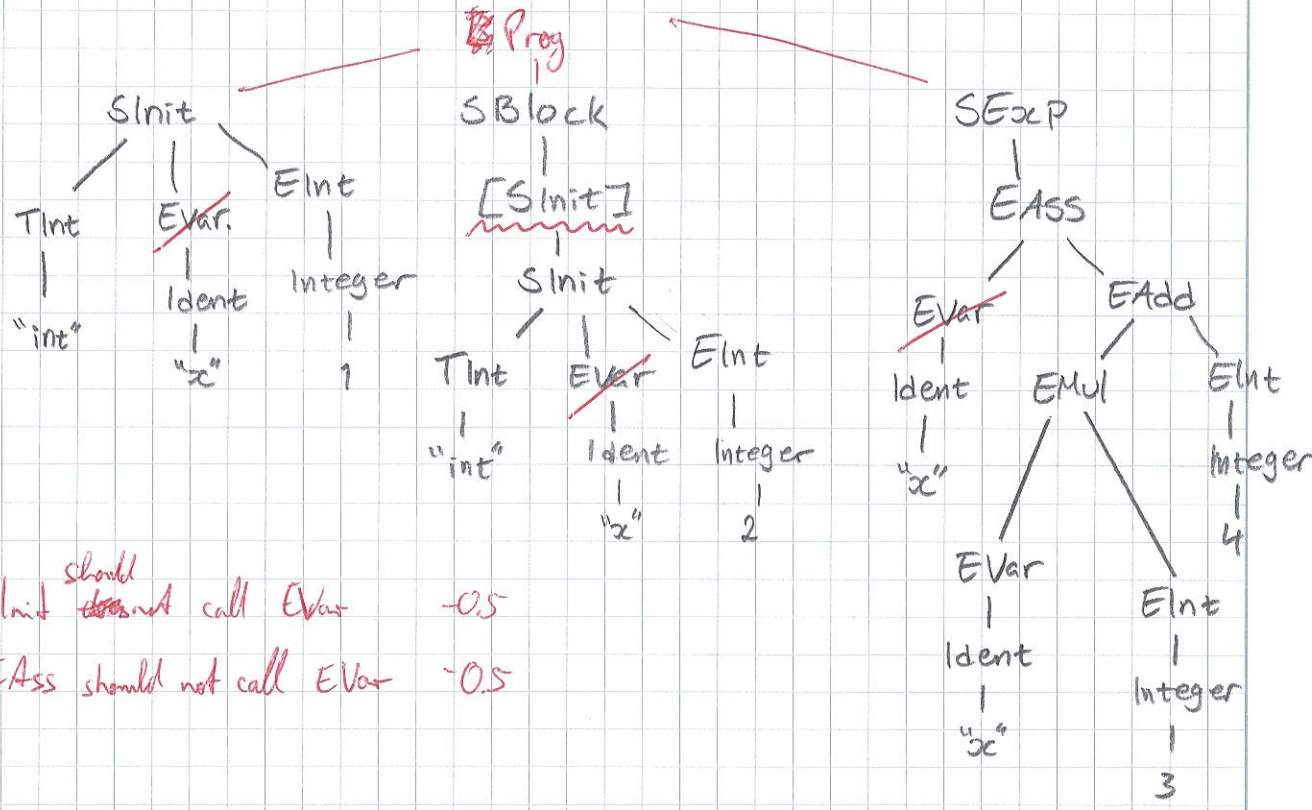
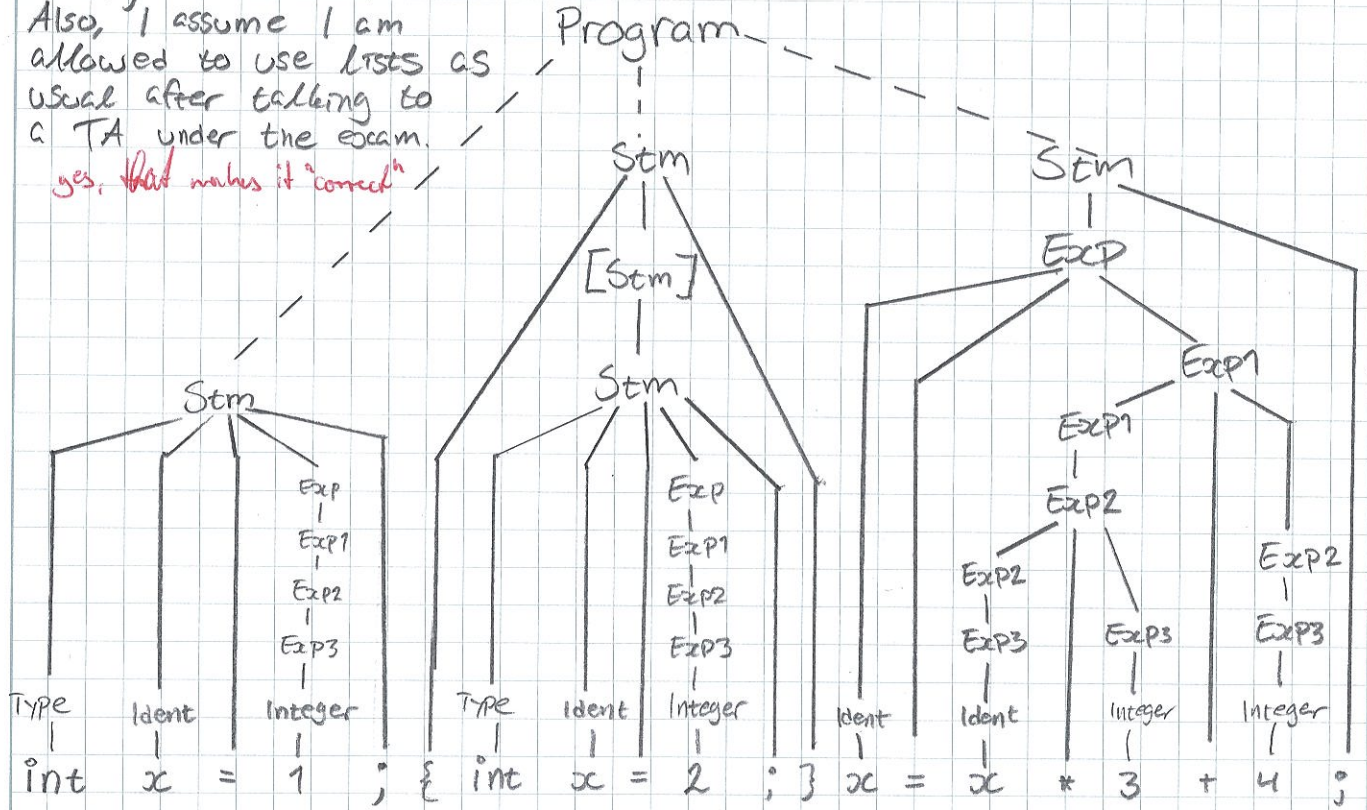
BNF grammar with:	Blocks	type int	Variables	
	Variable Inits Exp ";"		integer literals addition multiplication ASSIGNMENTS	
SBlock. SInit. SExp.	$Stm ::= \{ [Stm] \}$ $Stm ::= Type\ Ident\ "="\ Exp\ ";"$ $Stm ::= Exp\ ";"$; / / / /	-2
TInt.	Type ::= "int"		; /	
EVar. EInt. EAdd. EMul. EAss.	$Exp3 ::= Ident$ $Exp3 ::= Integer$ $Exp1 ::= Exp1\ "+" \ Exp2$ $Exp2 ::= Exp2\ "*" \ Exp3$ $Exp ::= Ident\ "=" \ Exp$; / / / / / /	
COERCIONS	Exp3	does not allow $x=y=42$, only $x=(y=42)$;	-1

I do not know how to define lists, why I break the rules in SBlock.
at least you are aware of it...

DAT151-3

5+4=9

The Program is not defined, however 3 dangling statements doesn't really make sense.
 Also, I assume I am allowed to use lists as usual after talking to a TA under the exam.
 yes, that makes it "correct"



Sinit should not call EVar -0.5
 EAss should not call EVar -0.5

Type check of Statements

SBlock.
$$\frac{\Gamma \vdash S_1 \dots S_m \text{ valid} \quad \Gamma \vdash S_2 \dots S_n \text{ valid}}{\Gamma \vdash \{S_1 \dots S_m\} S_2 \dots S_n \text{ valid}}$$

should add empty env for $\Gamma_1, \dots, \Gamma_n$ -0.5

SInit
$$\frac{\Gamma \vdash e : t}{\Gamma \vdash t \text{ id} = e; \text{ valid}}$$

should rule out that id is already in Γ -0.5

SExp
$$\frac{\Gamma \vdash e : t}{\Gamma \vdash e; \text{ valid}}$$

- cases $t \text{ id} = e; ss$ and $e; ss$ not -1
- scoping handling unclear -0.5

Interpretation of expressions

EVar
$$\frac{}{\Gamma \vdash \text{id} \Downarrow \langle v, \Gamma \rangle}$$
 id has value v in Γ

EInt
$$\frac{}{\Gamma \vdash i \Downarrow \langle i, \Gamma \rangle}$$

EAdd
$$\frac{\Gamma \vdash e_1 \Downarrow \langle v_1, \Gamma^1 \rangle \quad \Gamma^1 \vdash e_2 \Downarrow \langle v_2, \Gamma^2 \rangle}{\Gamma \vdash e_1 + e_2 \Downarrow \langle v_1 + v_2, \Gamma^2 \rangle}$$

EMul
$$\frac{\Gamma \vdash e_1 \Downarrow \langle v_1, \Gamma^1 \rangle \quad \Gamma^1 \vdash e_2 \Downarrow \langle v_2, \Gamma^2 \rangle}{\Gamma \vdash e_1 * e_2 \Downarrow \langle v_1 * v_2, \Gamma^2 \rangle}$$

EAss
$$\frac{\Gamma^1 \vdash e \Downarrow \langle v, \Gamma^1 \rangle}{\Gamma \vdash \text{id} = e \Downarrow \langle v, \Gamma \rangle (\text{id} = v)}$$

$x = x * 3 + 4$

- $x = x * 3 + 4$ S ✓
- $x = x * 3 + 4$ R ✓
- $Exp3 = x * 3 + 4$ S ✓
- $Exp3 = x * 3 + 4$ S ✓
- $Exp3 = x * 3 + 4$ R ✓
- $Exp3 = Exp3 * 3 + 4$ R ✓
- $Exp3 = Exp2 * 3 + 4$ S ✓
- $Exp3 = Exp2 * 3 + 4$ S ✓
- $Exp3 = Exp2 * 3 + 4$ R ✓
- $Exp3 = Exp2 * Exp3 + 4$ R ✓
- $Exp3 = Exp2 + 4$ R ✓
- $Exp3 = Exp1 + 4$ S ✓
- $Exp3 = Exp1 + 4$ S ✓
- $Exp3 = Exp1 + 4$ R ✓
- $Exp3 = Exp1 + Exp3$ R ✓
- $Exp3 = Exp1 + Exp2$ R ✓
- $Exp3 = Exp1$ R ✓
- Exp ~~R~~ Accept

← gets you into a dead end -1

Rule is $Idat = Exp1 \rightarrow Exp -1$

Compile SInt id e
 a = create id
 compile e 1
 emit(istore a)

Compile SBlock s
 newBlock
 for r in s:
 compile r 1
 remBlock

Compile SExp e
 compile e 1
 emit(pop)

Compile EVar id
 a = lookup id 1
 emit(istore a)

Compile EInt i 0,5
 emit(ldc i)

Compile EAdd e1 e2
 compile e1
 compile e2 0,25
 emit(iadd)

Compile EMul e1 e2
 compile e1
 compile e2 0,25
 emit(imul)

Compile EAss id e
 a = lookup id
 compile e 0,5
 emit(istore a)
 Load

istore a: $\langle P, V, S.v \rangle \rightarrow \langle P+1, V(a=v), S \rangle$
 iload a: $\langle P, V(a=v), S \rangle \rightarrow \langle P+1, V(a=v), S.v \rangle$
 iadd : $\langle P, V, S.v.w \rangle \rightarrow \langle P+1, V, S.(v+w) \rangle$
 imul : $\langle P, V, S.v.w \rangle \rightarrow \langle P+1, V, S.(v*w) \rangle$
 ldc i: $\langle P, V, S \rangle \rightarrow \langle P+1, V, S.i \rangle$
 pop : $\langle P, V, S.x \rangle \rightarrow \langle P+1, V, S \rangle$ 6

Sorry about the emit-mess I forget them at first... :(

Nothing compared to other exams, :p.