Seminar 10 week 10 (4 December 2023 – 9 December 2023)

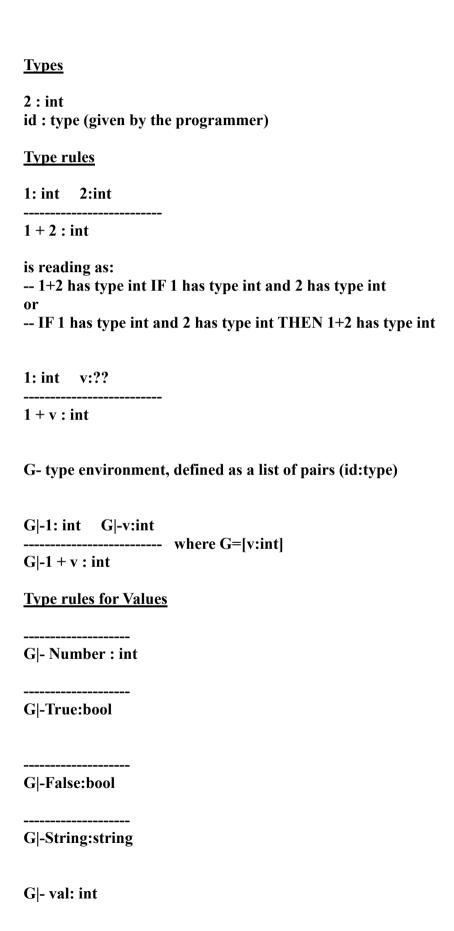
1. Discussion of what is a type checker.

Toy language syntax:

```
Type ::= int
bool
string
| Ref Type
void
Stmt ::= Stmt;Stmt
  | Id= Exp
  | Type Id
  | print(Exp)
  | If Exp Then Stmt1 else Stm2
  | Nop
  openRFile(Exp)
  | readFile(Exp, id)
  | closeRFile(Exp)
  new(Id, Exp)
 | wH(Id, Exp)
 | while Exp Stmt
 | fork(Stmt)
Value ::= Number
 True
 |False
 String
 | (value, Type) //ref value
Exp ::= Value
 | id
 | rH(Exp)
 | Exp1 + Exp2 |
 | Exp1 - Exp2
 | Exp1 * Exp2
 | Exp1 / Exp2
 | Exp1 and Exp2
 | Exp1 or Exp2
 | Exp1 < Exp2 |
 | Exp1 \le Exp2
 | Exp1 == Exp2
```

| Exp1 != Exp2

```
| Exp1 > Exp2
| Exp1 >= Exp2
```



G|-(val,type) : Ref type

Type rules for Expressions

(id:t) is in G

G|- id:t

G|- e1: int G|-e2:int

G|-e1 + e2:int

the same rule for -,*,/

G|-e1: bool G|-e2:bool

G|-e1 and e2:bool the same rule for or

G|- e1: int G|-e2:int

G|-e1 < e2:bool

the same rule for <=,==,!=,>, >=

G|- **e**1 :**Ref t**1

G|- **r**H(**e**1): **t**1

Type rules for Statements

G|- **s**1:void,**G**1 **G**1|- **s**2:void,**G**2

G|- s1;s2: void,G2

G|-id:t1 G|-exp:t2 t1==t2

G|- id=exp: void,G

G|- type id : void, G+[(id:type)] G|- exp:t G|- print(exp): void,G G|- e: bool G|-s1:void,G1G|-s2:void,G2G|- if e then s1 else s2 : void,G G|- nop:void, G G|- exp:string G|-id:int G|- readFile(exp,id):void,G G|- exp:string G|- openRFile(exp):void, G G|- exp:string G|- closeRFile(exp):void, G G|- exp:t G|- id:Ref t G|- new(id,exp):void, G G|- exp:t G|- id:Ref tG|- wH(id,exp):void, G G|- exp:bool G|- stmt:void,G1

G|- while exp stmt:void, G

G - stmt:void,G1
G - fork(stmt):void, G

2. Discussion of the assignment A6. The deadline of the assignment A6 is week 12 (18 - 22 December 2023).