school of | Computer Science & Statistics

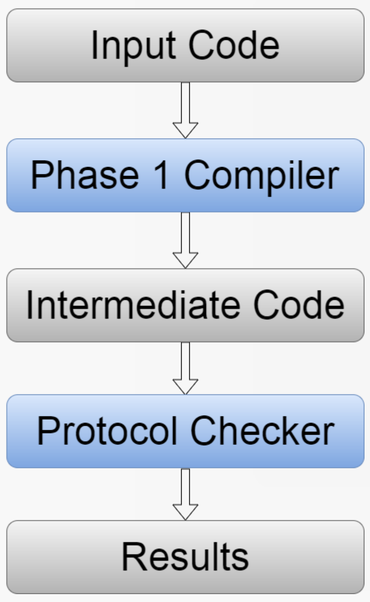
Formal Verification of Session Types

Student: Théa Johnson

Supervisor: Vasileios Koutavas

Project Outline

Aim: To implement a protocol checker for behaviours in a system to verify communication over a distributed system



Dig. Code Progression

My Contributions:

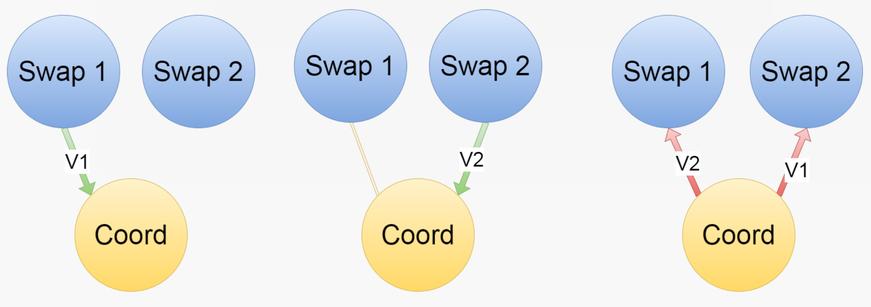
- A text based syntax for intermediate code
- Parser and Lexical Analyser combination to translate the intermediate code
- An implementation of the protocol checker design

Input Code

Two swap functions that exchange Values through the co-ordinator

```
let fun coord(_) =
  let val p1 = acc-swp ()
    val x1 = recv p1
    val p2 = acc-swp ()
    val x2 = recv p2
  in send p2 x1; send p1 x2; coord ()
in spawn coord;

let fun swap(x) =
  let val p = req-swp ()
  in send p x; recv p
in spawn (fn _ => swap 1);
    spawn (fn _ => swap 2);
```



Intermediate Code

Behaviours:

```
tau;
tau;
spn(B64);
tau;
tau;
spn(tau; tau; tau;
tau; B122; tau; tau;
tau; B123; tau; tau;
B124; tau);
tau;
spn(tau; tau; tau;
tau; B138; tau; tau;
tau; B139; tau; tau;
B140; tau)
```

Region Constraints:

```
R77 ~ $l1$,
R80 ~ $l1$,
R78 ~ $l2$,
R79 ~ $l2$,
R125 ~ $l3$,
R126 ~ $l3$,
R141 ~ $l3$,
R142 ~ $l3$,
R77 ~ R12,
```

Behaviour Constraints:

```
rec B64 (tau; tau; B71; tau;
tau; B72; tau; tau; B73; tau;
tau; B74; tau; tau; tau; B75;
tau; tau; tau; B76; tau; tau;
B64) < B64,
psh($l1$,? int ! int end) < B71,
R77 ? int < B72,
psh($l2$,? int ! int end) < B73,
R78 ? int < B74,
R79 ! int < B75,
R80 ! int < B76,

psh($l3$,! int ? int end) <
B122,
R125 ! int < B123,
R126 ? int < B124,

psh($l3$,! int ? int end) <
B138,
R141 ! int < B139,
R142 ? int < B140,
```

Channel Constraints:

```
Cswapl1` ~ ? int ! int end,
Cswapl2` ~ ? int ! int end,
Cswapl3` ~ ! int ? int end,
```

Output

B.A.(Mod.) Computer Science