Programming Paradigms Final Project: Test Report

Group 26
Martijn Verkleij & Tim Kerkhoven

University of Twente
m.f.verkleij@student.utwente.nl, t.kerkhoven@student.utwente.nl
s1466895 s1375253

June 26, 2017

Contents

Chapter 1

Syntax Tests

Syntax 1

Source

```
int number;

procedure p() {}
```

Output

```
Main: tokenList not fully parsed
```

Syntax 2

Source

```
procedure(j, int i) {};
```

Output

```
Main: tokenList not fully parsed
```

Syntax 3

Source

```
int 5num = 5;
```

Output

```
Main: tokenList not fully parsed
```

Syntax 4

Source

```
1 if (true) else {// do nothing}
```

Output

```
1 Main: tokenList not fully parsed
```

Syntax 5

Source

```
1 print(5*(3-1));
```

Output

```
1 Main: tokenList not fully parsed
```

Chapter 2

Contextual Tests

Wrong Type

Source

```
int i = 1;
if (i) {
    print(i);
}
```

Output

```
Main: Condition in if statement should be of type: bool, but isnt, in: ASTVar "i" ([],[],[])
```

Not Declared

Source

```
if (i) {
    print(1);
}
```

Output

```
Main: Variable: i not declared in Checker.getExprType.iterVar
```

Chapter 3

Semantic Tests

Banking

Source

```
global int john = 10000;
global int jane = 2000;
global int martijn = 99999;

procedure deposit(int account, int amount) {
    account = (account + amount);
}

procedure withdraw(int account, int amount) {
    if ((account >= amount)) {
        account = (account - amount);
    }
}

procedure transfer(int sender, int target, int amount) {
    if ((sender >= amount)) {
        sender = (sender - amount);
        target = (target + amount);
    }
}

procedure test1() {
    print(john, jane, martijn);
    fork deposit(jane, 100);
    fork deposit(john, 100);
    fork deposit(martijn, 1);
    join;
    print(john, jane, martijn);

    fork deposit(jane, 200);
```

```
31      fork withdraw(john, 200);
32      fork deposit(martijn, 2000);
33      join;
34      print(john, jane, martijn);
35      fork withdraw(jane, 10);
36      fork withdraw(john, 20);
37      fork deposit(martijn, 100);
38      join;
39      print(john, jane, martijn);
40      fork withdraw(jane, 300);
41      fork withdraw(john, 30000);
42      fork withdraw(martijn, 50);
43      join;
44      print(john, jane, martijn);
45      fork withdraw(jane, 35);
46      fork transfer(martijn, john, 1000);
47      join;
48      print(john, jane, martijn);
49      fork transfer(martijn, jane, 100);
50      join;
51      print(john, jane, martijn);
52
53  }
54
55  test1();
```

Generated SprIL

```
0   Branch 1 (Rel 6)
1   TestAndSet (DirAddr 2)
2   Receive 6
3   Branch 6 (Rel 2)
4   Jump (Rel (-3))
5   Jump (Rel 1470)
6   ReadInstr (DirAddr 0)
7   Receive 3
8   Compute Equal 3 0 6
9   Branch 6 (Rel 2)
10  EndProg
11  TestAndSet (DirAddr 2)
12  Receive 6
13  Branch 6 (Rel 2)
14  Jump (Rel (-8))
15  ComputeI Add 1 30 3
16  TestAndSet (IndAddr 3)
17  Receive 6
18  Branch 6 (Rel 2)
19  Jump (Rel (-3))
```

```
ReadInstr (DirAddr 3)
Receive 3
Push 3
ComputeI Add 7 1 4
ReadInstr (DirAddr 4)
Receive 5
Load (ImmValue 5) 2
Compute Equal 5 0 6
Branch 6 (Rel 18)
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
Compute Decr 5 0 5
Jump (Rel (-18))
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Load (ImmValue 7) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
ComputeI Gt 5 2 6
Branch 6 (Rel 7)
Load (IndAddr 2) 3
Compute Add 7 5 6
Store 3 (IndAddr 6)
Compute Incr 5 0 5
ComputeI Add 2 3 2
Jump (Rel (-7))
```

```
Compute Add 7 0 4
ComputeI Add 4 3 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 2 6
Load (IndAddr 6) 5
Push 5
Pop 3
Pop 2
Compute Add 2 3 4
Push 4
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Pop 2
Store 2 (IndAddr 6)
Push 2
Pop 0
Load (IndAddr 7) 7
Load (ImmValue 6) 2
Compute Sub 7 2 2
ComputeI Add 0 1 5
ComputeI Gt 5 2 6
Branch 6 (Rel 23)
Compute Add 7 5 6
Load (IndAddr 6) 4
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 2)
Store 4 (IndAddr 3)
Compute Incr 2 0 2
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 10)
Compute Add 3 0 6
TestAndSet (IndAddr 6)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-4))
ComputeI Add 3 1 3
WriteInstr 4 (IndAddr 3)
```

```
116 ComputeI Sub 3 1 3
117 WriteInstr 0 (IndAddr 3)
118 Compute Incr 5 0 5
119 ComputeI Add 2 2 2
120 Jump (Rel (-23))
121 Compute Decr 7 0 2
122 Load (IndAddr 2) 6
123 Load (IndAddr 7) 7
124 Jump (Ind 6)
125 Load (ImmValue 7) 2
126 Compute Sub 7 2 2
127 Load (ImmValue 1) 5
128 ComputeI Gt 5 2 6
129 Branch 6 (Rel 7)
130 Load (IndAddr 2) 3
131 Compute Add 7 5 6
132 Store 3 (IndAddr 6)
133 Compute Incr 5 0 5
134 ComputeI Add 2 3 2
135 Jump (Rel (-7))
136 Compute Add 7 0 4
137 ComputeI Add 4 3 4
138 Store 7 (IndAddr 4)
139 Compute Add 4 0 7
140 Compute Add 7 0 6
141 Load (IndAddr 6) 6
142 ComputeI Add 6 1 6
143 Load (IndAddr 6) 5
144 Push 5
145 Compute Add 7 0 6
146 Load (IndAddr 6) 6
147 ComputeI Add 6 2 6
148 Load (IndAddr 6) 5
149 Push 5
150 Pop 3
151 Pop 2
152 Compute GtE 2 3 4
153 Push 4
154 Pop 6
155 ComputeI Xor 6 1 6
156 Branch 6 (Rel 30)
157 Compute Add 7 0 4
158 ComputeI Add 4 1 4
159 Store 7 (IndAddr 4)
160 Compute Add 4 0 7
161 Compute Add 7 0 6
162 Load (IndAddr 6) 6
163 Load (IndAddr 6) 6
```

```
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 2 6
Load (IndAddr 6) 5
Push 5
Pop 3
Pop 2
Compute Sub 2 3 4
Push 4
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Pop 2
Store 2 (IndAddr 6)
Push 2
Pop 0
Load (IndAddr 7) 7
Load (IndAddr 7) 7
Load (ImmValue 6) 2
Compute Sub 7 2 2
ComputeI Add 0 1 5
ComputeI Gt 5 2 6
Branch 6 (Rel 23)
Compute Add 7 5 6
Load (IndAddr 6) 4
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 2)
Store 4 (IndAddr 3)
Compute Incr 2 0 2
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 10)
Compute Add 3 0 6
TestAndSet (IndAddr 6)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-4))
ComputeI Add 3 1 3
WriteInstr 4 (IndAddr 3)
ComputeI Sub 3 1 3
WriteInstr 0 (IndAddr 3)
Compute Incr 5 0 5
```

```
212 ComputeI Add 2 2 2
213 Jump (Rel (-23))
214 Compute Decr 7 0 2
215 Load (IndAddr 2) 6
216 Load (IndAddr 7) 7
217 Jump (Ind 6)
218 Load (ImmValue 10) 2
219 Compute Sub 7 2 2
220 Load (ImmValue 1) 5
221 ComputeI Gt 5 3 6
222 Branch 6 (Rel 7)
223 Load (IndAddr 2) 3
224 Compute Add 7 5 6
225 Store 3 (IndAddr 6)
226 Compute Incr 5 0 5
227 ComputeI Add 2 3 2
228 Jump (Rel (-7))
229 Compute Add 7 0 4
230 ComputeI Add 4 4 4
231 Store 7 (IndAddr 4)
232 Compute Add 4 0 7
233 Compute Add 7 0 6
234 Load (IndAddr 6) 6
235 ComputeI Add 6 1 6
236 Load (IndAddr 6) 5
237 Push 5
238 Compute Add 7 0 6
239 Load (IndAddr 6) 6
240 ComputeI Add 6 3 6
241 Load (IndAddr 6) 5
242 Push 5
243 Pop 3
244 Pop 2
245 Compute GtE 2 3 4
246 Push 4
247 Pop 6
248 ComputeI Xor 6 1 6
249 Branch 6 (Rel 54)
250 Compute Add 7 0 4
251 ComputeI Add 4 1 4
252 Store 7 (IndAddr 4)
253 Compute Add 4 0 7
254 Compute Add 7 0 6
255 Load (IndAddr 6) 6
256 Load (IndAddr 6) 6
257 ComputeI Add 6 1 6
258 Load (IndAddr 6) 5
259 Push 5
```

```
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 3 6
Load (IndAddr 6) 5
Push 5
Pop 3
Pop 2
Compute Sub 2 3 4
Push 4
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Pop 2
Store 2 (IndAddr 6)
Push 2
Pop 0
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 2 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 3 6
Load (IndAddr 6) 5
Push 5
Pop 3
Pop 2
Compute Add 2 3 4
Push 4
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 2 6
Pop 2
Store 2 (IndAddr 6)
Push 2
Pop 0
Load (IndAddr 7) 7
Load (IndAddr 7) 7
Load (ImmValue 9) 2
Compute Sub 7 2 2
ComputeI Add 0 1 5
ComputeI Gt 5 3 6
```

```
Branch 6 (Rel 23)
Compute Add 7 5 6
Load (IndAddr 6) 4
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 2)
Store 4 (IndAddr 3)
Compute Incr 2 0 2
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 10)
Compute Add 3 0 6
TestAndSet (IndAddr 6)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-4))
ComputeI Add 3 1 3
WriteInstr 4 (IndAddr 3)
ComputeI Sub 3 1 3
WriteInstr 0 (IndAddr 3)
Compute Incr 5 0 5
ComputeI Add 2 2 2
Jump (Rel (-23))
Compute Decr 7 0 2
Load (IndAddr 2) 6
Load (IndAddr 7) 7
Jump (Ind 6)
Load (ImmValue 1) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
ComputeI Gt 5 0 6
Branch 6 (Rel 7)
Load (IndAddr 2) 3
Compute Add 7 5 6
Store 3 (IndAddr 6)
Compute Incr 5 0 5
ComputeI Add 2 3 2
Jump (Rel (-7))
Compute Add 7 0 4
ComputeI Add 4 1 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
```

```
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 100) 6
Push 6
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 5) 4
Pop 3
```

```
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 33) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 57) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 100) 6
Push 6
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 5) 4
Pop 3
WriteInstr 3 (IndAddr 4)
```

```
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 35) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 57) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 1) 6
Push 6
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 5) 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
```

```
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 37) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 57) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Compute Equal 0 1 6
Branch 6 (Rel 4)
Load (ImmValue 2) 2
PrintOut 2
EndProg
Load (ImmValue 30) 3
Load (ImmValue 0) 2
ReadInstr (IndAddr 3)
Receive 4
Compute Add 2 4 2
ComputeI NEq 3 33 6
Compute Incr 3 0 3
Branch 6 (Rel (-5))
Compute Equal 2 0 6
Branch 6 (Rel 2)
Jump (Rel (-10))
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
```

```
548 Load (ImmValue 38) 4
549 ReadInstr (IndAddr 4)
550 Receive 5
551 Push 5
552 WriteInstr 0 (IndAddr 2)
553 Load (ImmValue 33) 2
554 TestAndSet (IndAddr 2)
555 Receive 3
556 Branch 3 (Rel 2)
557 Jump (Rel (-4))
558 Load (ImmValue 34) 4
559 ReadInstr (IndAddr 4)
560 Receive 5
561 Push 5
562 WriteInstr 0 (IndAddr 2)
563 Load (ImmValue 35) 2
564 TestAndSet (IndAddr 2)
565 Receive 3
566 Branch 3 (Rel 2)
567 Jump (Rel (-4))
568 Load (ImmValue 36) 4
569 ReadInstr (IndAddr 4)
570 Receive 5
571 Push 5
572 WriteInstr 0 (IndAddr 2)
573 Pop 6
574 PrintOut 6
575 Pop 6
576 PrintOut 6
577 Pop 6
578 PrintOut 6
579 TestAndSet (DirAddr 1)
580 Receive 6
581 Branch 6 (Rel 2)
582 Jump (Rel (-3))
583 Load (ImmValue 200) 6
584 Push 6
585 Load (ImmValue 33) 2
586 TestAndSet (IndAddr 2)
587 Receive 3
588 Branch 3 (Rel 2)
589 Jump (Rel (-4))
590 Load (ImmValue 34) 4
591 ReadInstr (IndAddr 4)
592 Receive 5
593 Push 5
594 WriteInstr 0 (IndAddr 2)
595 Load (ImmValue 5) 4
```

```
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 33) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 57) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 200) 6
Push 6
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 5) 4
Pop 3
```

```
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 35) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 125) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 2000) 6
Push 6
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 5) 4
Pop 3
WriteInstr 3 (IndAddr 4)
```

```
692 Compute Incr 4 0 4
693 Load (ImmValue (-1)) 3
694 WriteInstr 3 (IndAddr 4)
695 Compute Incr 4 0 4
696 Load (ImmValue 37) 3
697 WriteInstr 3 (IndAddr 4)
698 Compute Incr 4 0 4
699 Pop 3
700 WriteInstr 3 (IndAddr 4)
701 Compute Incr 4 0 4
702 Load (ImmValue (-1)) 3
703 WriteInstr 3 (IndAddr 4)
704 Compute Incr 4 0 4
705 Load (ImmValue (-1)) 3
706 WriteInstr 3 (IndAddr 4)
707 Compute Incr 4 0 4
708 Load (ImmValue 2) 5
709 WriteInstr 5 (DirAddr 4)
710 Load (ImmValue 57) 6
711 Push 6
712 Pop 5
713 WriteInstr 5 (DirAddr 3)
714 WriteInstr 0 (DirAddr 2)
715 Load (ImmValue 1) 3
716 ReadInstr (IndAddr 3)
717 Receive 6
718 Branch 6 (Rel 2)
719 Jump (Rel (-3))
720 Compute Equal 0 1 6
721 Branch 6 (Rel 4)
722 Load (ImmValue 2) 2
723 PrintOut 2
724 EndProg
725 Load (ImmValue 30) 3
726 Load (ImmValue 0) 2
727 ReadInstr (IndAddr 3)
728 Receive 4
729 Compute Add 2 4 2
730 ComputeI NEq 3 33 6
731 Compute Incr 3 0 3
732 Branch 6 (Rel (-5))
733 Compute Equal 2 0 6
734 Branch 6 (Rel 2)
735 Jump (Rel (-10))
736 Load (ImmValue 37) 2
737 TestAndSet (IndAddr 2)
738 Receive 3
739 Branch 3 (Rel 2)
```

```
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 10) 6
Push 6
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
```

```
Load (ImmValue 5) 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 33) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 125) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 20) 6
Push 6
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 5) 4
```

```
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 35) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 125) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 100) 6
Push 6
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 5) 4
Pop 3
```

```
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 37) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 57) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Compute Equal 0 1 6
Branch 6 (Rel 4)
Load (ImmValue 2) 2
PrintOut 2
EndProg
Load (ImmValue 30) 3
Load (ImmValue 0) 2
ReadInstr (IndAddr 3)
Receive 4
Compute Add 2 4 2
ComputeI NEq 3 33 6
Compute Incr 3 0 3
Branch 6 (Rel (-5))
Compute Equal 2 0 6
Branch 6 (Rel 2)
Jump (Rel (-10))
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
```

```
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 300) 6
Push 6
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
```

```
WriteInstr 0 (IndAddr 2)
Load (ImmValue 5) 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 33) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 125) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 30000) 6
Push 6
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
```

```
Load (ImmValue 5) 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 35) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 125) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 50) 6
Push 6
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 5) 4
```

```
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 37) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 125) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Compute Equal 0 1 6
Branch 6 (Rel 4)
Load (ImmValue 2) 2
PrintOut 2
EndProg
Load (ImmValue 30) 3
Load (ImmValue 0) 2
ReadInstr (IndAddr 3)
Receive 4
Compute Add 2 4 2
ComputeI NEq 3 33 6
Compute Incr 3 0 3
Branch 6 (Rel (-5))
Compute Equal 2 0 6
Branch 6 (Rel 2)
Jump (Rel (-10))
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
```

```
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 35) 6
Push 6
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
```

```
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 5) 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 33) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 125) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 1000) 6
Push 6
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
```

```
WriteInstr 0 (IndAddr 2)
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 5) 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 37) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 35) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 3) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 218) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
```

```
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Compute Equal 0 1 6
Branch 6 (Rel 4)
Load (ImmValue 2) 2
PrintOut 2
EndProg
Load (ImmValue 30) 3
Load (ImmValue 0) 2
ReadInstr (IndAddr 3)
Receive 4
Compute Add 2 4 2
ComputeI NEq 3 33 6
Compute Incr 3 0 3
Branch 6 (Rel (-5))
Compute Equal 2 0 6
Branch 6 (Rel 2)
Jump (Rel (-10))
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
```

```
WriteInstr 0 (IndAddr 2)
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 100) 6
Push 6
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 5) 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 37) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
```

```
Compute Incr 4 0 4
Load (ImmValue 33) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
WriteInstr 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 3) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 218) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Compute Equal 0 1 6
Branch 6 (Rel 4)
Load (ImmValue 2) 2
PrintOut 2
EndProg
Load (ImmValue 30) 3
Load (ImmValue 0) 2
ReadInstr (IndAddr 3)
Receive 4
Compute Add 2 4 2
ComputeI NEq 3 33 6
Compute Incr 3 0 3
Branch 6 (Rel (-5))
Compute Equal 2 0 6
Branch 6 (Rel 2)
Jump (Rel (-10))
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
```

```
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Load (IndAddr 7) 7
Load (ImmValue 0) 2
Compute Sub 7 2 2
ComputeI Add 0 1 5
ComputeI Gt 5 0 6
Branch 6 (Rel 23)
Compute Add 7 5 6
Load (IndAddr 6) 4
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 2)
Store 4 (IndAddr 3)
Compute Incr 2 0 2
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 10)
Compute Add 3 0 6
TestAndSet (IndAddr 6)
Receive 6
```

```
Branch 6 (Rel 2)
Jump (Rel (-4))
ComputeI Add 3 1 3
WriteInstr 4 (IndAddr 3)
ComputeI Sub 3 1 3
WriteInstr 0 (IndAddr 3)
Compute Incr 5 0 5
ComputeI Add 2 2 2
Jump (Rel (-23))
Compute Decr 7 0 2
Load (IndAddr 2) 6
Load (IndAddr 7) 7
Jump (Ind 6)
Nop
Nop
Load (ImmValue 10000) 6
Push 6
Pop 6
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 36) 4
WriteInstr 6 (IndAddr 4)
WriteInstr 0 (IndAddr 2)
Load (ImmValue 2000) 6
Push 6
Pop 6
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 34) 4
WriteInstr 6 (IndAddr 4)
WriteInstr 0 (IndAddr 2)
Load (ImmValue 99999) 6
Push 6
Pop 6
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 38) 4
WriteInstr 6 (IndAddr 4)
WriteInstr 0 (IndAddr 2)
```

```
Compute Add 7 0 4
ComputeI Add 4 1 4
Load (ImmValue 0) 5
Load (ImmValue 1522) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 335) 6
Push 6
Pop 2
Jump (Ind 2)
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 10000
>>> 2000
>>> 99999
>>> 10100
>>> 2100
>>> 100000
>>> 9900
>>> 2300
>>> 102000
>>> 9880
>>> 2290
>>> 102100
>>> 9880
>>> 1990
>>> 102050
>>> 10880
>>> 1955
>>> 101050
>>> 10880
>>> 2055
>>> 100950
```

Blocks

Source

```
int x = 1;
int y = 100;
```

```
bool a = false;
bool b = false;
print(x,y,a,b);
{
    int x = 2;
    int y = 120;
    bool a = true;
    bool b = false;
    print(x,y,a,b);
    {   {}
        int x = 3;
        int y = 123;
        bool a = false;
        bool b = true;
        print(x,y,a,b);
        {
            int x = 4;
            int y = 423;
            bool a = true;
            bool b = true;
            print(x,y,a,b);
        }
        print(x,y,a,b);
        {
            int x = 5;
            int y = 453;
            bool a = true;
            bool b = false;
            print(x,y,a,b);
            {
                int x = 5;
                int y = 453;
                bool a = false;
                bool b = true;
                print(x,y,a,b);
            }
            print(x,y,a,b);
            {
                int x = 6;
                int y = 456;
                bool a = false;
                bool b = false;
                print(x,y,a,b);
            }
            print(x,y,a,b);
        }
        print(x,y,a,b);
    }
```

```
51      print(x,y,a,b);
52      {
53          print(x,y,a,b);
54          {
55              print(x,y,a,b);
56          }
57      }
58      print(x,y,a,b);
59  }
60  print(x,y,a,b);
```

Generated SprIL

```
0   Branch 1 (Rel 6)
1   TestAndSet (DirAddr 2)
2   Receive 6
3   Branch 6 (Rel 2)
4   Jump (Rel (-3))
5   Jump (Rel 54)
6   ReadInstr (DirAddr 0)
7   Receive 3
8   Compute Equal 3 0 6
9   Branch 6 (Rel 2)
10  EndProg
11  TestAndSet (DirAddr 2)
12  Receive 6
13  Branch 6 (Rel 2)
14  Jump (Rel (-8))
15  ComputeI Add 1 30 3
16  TestAndSet (IndAddr 3)
17  Receive 6
18  Branch 6 (Rel 2)
19  Jump (Rel (-3))
20  ReadInstr (DirAddr 3)
21  Receive 3
22  Push 3
23  ComputeI Add 7 1 4
24  ReadInstr (DirAddr 4)
25  Receive 5
26  Load (ImmValue 5) 2
27  Compute Equal 5 0 6
28  Branch 6 (Rel 18)
29  ReadInstr (IndAddr 2)
30  Receive 3
31  Store 3 (IndAddr 4)
32  Compute Incr 2 0 2
33  Compute Incr 4 0 4
34  ReadInstr (IndAddr 2)
```

```
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
Compute Decr 5 0 5
Jump (Rel (-18))
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Nop
Nop
Load (ImmValue 1) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 1 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 100) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 2 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 0) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 3 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 0) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 4 6
Pop 5
Store 5 (IndAddr 6)
```

```
83  Compute Add 7 0 6
84  ComputeI Add 6 4 6
85  Load (IndAddr 6) 5
86  Push 5
87  Compute Add 7 0 6
88  ComputeI Add 6 3 6
89  Load (IndAddr 6) 5
90  Push 5
91  Compute Add 7 0 6
92  ComputeI Add 6 2 6
93  Load (IndAddr 6) 5
94  Push 5
95  Compute Add 7 0 6
96  ComputeI Add 6 1 6
97  Load (IndAddr 6) 5
98  Push 5
99  Pop 6
100 PrintOut 6
101 Pop 6
102 PrintOut 6
103 Pop 6
104 PrintOut 6
105 Pop 6
106 PrintOut 6
107 Compute Add 7 0 4
108 ComputeI Add 4 5 4
109 Store 7 (IndAddr 4)
110 Compute Add 4 0 7
111 Load (ImmValue 2) 6
112 Push 6
113 Compute Add 7 0 6
114 ComputeI Add 6 1 6
115 Pop 5
116 Store 5 (IndAddr 6)
117 Load (ImmValue 120) 6
118 Push 6
119 Compute Add 7 0 6
120 ComputeI Add 6 2 6
121 Pop 5
122 Store 5 (IndAddr 6)
123 Load (ImmValue 1) 6
124 Push 6
125 Compute Add 7 0 6
126 ComputeI Add 6 3 6
127 Pop 5
128 Store 5 (IndAddr 6)
129 Load (ImmValue 0) 6
130 Push 6
```

```
131 Compute Add 7 0 6
132 ComputeI Add 6 4 6
133 Pop 5
134 Store 5 (IndAddr 6)
135 Compute Add 7 0 6
136 ComputeI Add 6 4 6
137 Load (IndAddr 6) 5
138 Push 5
139 Compute Add 7 0 6
140 ComputeI Add 6 3 6
141 Load (IndAddr 6) 5
142 Push 5
143 Compute Add 7 0 6
144 ComputeI Add 6 2 6
145 Load (IndAddr 6) 5
146 Push 5
147 Compute Add 7 0 6
148 ComputeI Add 6 1 6
149 Load (IndAddr 6) 5
150 Push 5
151 Pop 6
152 PrintOut 6
153 Pop 6
154 PrintOut 6
155 Pop 6
156 PrintOut 6
157 Pop 6
158 PrintOut 6
159 Compute Add 7 0 4
160 ComputeI Add 4 5 4
161 Store 7 (IndAddr 4)
162 Compute Add 4 0 7
163 Compute Add 7 0 4
164 ComputeI Add 4 1 4
165 Store 7 (IndAddr 4)
166 Compute Add 4 0 7
167 Load (IndAddr 7) 7
168 Load (ImmValue 3) 6
169 Push 6
170 Compute Add 7 0 6
171 ComputeI Add 6 1 6
172 Pop 5
173 Store 5 (IndAddr 6)
174 Load (ImmValue 123) 6
175 Push 6
176 Compute Add 7 0 6
177 ComputeI Add 6 2 6
178 Pop 5
```

```
Store 5 (IndAddr 6)
Load (ImmValue 0) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 3 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 1) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 4 6
Pop 5
Store 5 (IndAddr 6)
Compute Add 7 0 6
ComputeI Add 6 4 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 3 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 2 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Compute Add 7 0 4
ComputeI Add 4 5 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 4) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 1 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 423) 6
```

```
227 Push 6
228 Compute Add 7 0 6
229 ComputeI Add 6 2 6
230 Pop 5
231 Store 5 (IndAddr 6)
232 Load (ImmValue 1) 6
233 Push 6
234 Compute Add 7 0 6
235 ComputeI Add 6 3 6
236 Pop 5
237 Store 5 (IndAddr 6)
238 Load (ImmValue 1) 6
239 Push 6
240 Compute Add 7 0 6
241 ComputeI Add 6 4 6
242 Pop 5
243 Store 5 (IndAddr 6)
244 Compute Add 7 0 6
245 ComputeI Add 6 4 6
246 Load (IndAddr 6) 5
247 Push 5
248 Compute Add 7 0 6
249 ComputeI Add 6 3 6
250 Load (IndAddr 6) 5
251 Push 5
252 Compute Add 7 0 6
253 ComputeI Add 6 2 6
254 Load (IndAddr 6) 5
255 Push 5
256 Compute Add 7 0 6
257 ComputeI Add 6 1 6
258 Load (IndAddr 6) 5
259 Push 5
260 Pop 6
261 PrintOut 6
262 Pop 6
263 PrintOut 6
264 Pop 6
265 PrintOut 6
266 Pop 6
267 PrintOut 6
268 Load (IndAddr 7) 7
269 Compute Add 7 0 6
270 ComputeI Add 6 4 6
271 Load (IndAddr 6) 5
272 Push 5
273 Compute Add 7 0 6
274 ComputeI Add 6 3 6
```

```
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 2 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Compute Add 7 0 4
ComputeI Add 4 5 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 5) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 1 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 453) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 2 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 1) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 3 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 0) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 4 6
Pop 5
Store 5 (IndAddr 6)
Compute Add 7 0 6
ComputeI Add 6 4 6
```

```
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 3 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 2 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Compute Add 7 0 4
ComputeI Add 4 5 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 5) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 1 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 453) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 2 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 0) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 3 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 1) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 4 6
```

```
Pop 5
Store 5 (IndAddr 6)
Compute Add 7 0 6
ComputeI Add 6 4 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 3 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 2 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Load (IndAddr 7) 7
Compute Add 7 0 6
ComputeI Add 6 4 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 3 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 2 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
```

```
PrintOut 6
Pop 6
PrintOut 6
Compute Add 7 0 4
ComputeI Add 4 5 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 6) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 1 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 456) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 2 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 0) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 3 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 0) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 4 6
Pop 5
Store 5 (IndAddr 6)
Compute Add 7 0 6
ComputeI Add 6 4 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 3 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 2 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
```

```
467 PrintOut 6
468 Pop 6
469 PrintOut 6
470 Pop 6
471 PrintOut 6
472 Pop 6
473 PrintOut 6
474 Load (IndAddr 7) 7
475 Compute Add 7 0 6
476 ComputeI Add 6 4 6
477 Load (IndAddr 6) 5
478 Push 5
479 Compute Add 7 0 6
480 ComputeI Add 6 3 6
481 Load (IndAddr 6) 5
482 Push 5
483 Compute Add 7 0 6
484 ComputeI Add 6 2 6
485 Load (IndAddr 6) 5
486 Push 5
487 Compute Add 7 0 6
488 ComputeI Add 6 1 6
489 Load (IndAddr 6) 5
490 Push 5
491 Pop 6
492 PrintOut 6
493 Pop 6
494 PrintOut 6
495 Pop 6
496 PrintOut 6
497 Pop 6
498 PrintOut 6
499 Load (IndAddr 7) 7
500 Compute Add 7 0 6
501 ComputeI Add 6 4 6
502 Load (IndAddr 6) 5
503 Push 5
504 Compute Add 7 0 6
505 ComputeI Add 6 3 6
506 Load (IndAddr 6) 5
507 Push 5
508 Compute Add 7 0 6
509 ComputeI Add 6 2 6
510 Load (IndAddr 6) 5
511 Push 5
512 Compute Add 7 0 6
513 ComputeI Add 6 1 6
514 Load (IndAddr 6) 5
```

```
Push 5
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
```

Results

```
>>> 1
>>> 100
>>> 0
>>> 0
>>> 2
>>> 120
>>> 1
>>> 0
>>> 3
>>> 123
>>> 0
>>> 1
>>> 4
>>> 423
>>> 1
>>> 1
>>> 3
>>> 123
>>> 0
>>> 1
>>> 5
>>> 453
>>> 1
>>> 0
>>> 5
>>> 453
>>> 0
>>> 1
>>> 5
>>> 453
>>> 1
>>> 0
>>> 6
>>> 456
>>> 0
>>> 0
>>> 5
>>> 453
>>> 1
```

```
>>> 0
>>> 3
>>> 123
>>> 0
>>> 1
>>> 2
>>> 120
>>> 1
>>> 0
>>> 2
>>> 120
>>> 1
>>> 0
>>> 2
>>> 120
>>> 1
>>> 0
>>> 2
>>> 120
>>> 1
>>> 0
>>> 1
>>> 100
>>> 0
>>> 0
```

Call-by-reference

Source

```
global int var = 1337;
global int y = 42;

procedure write(int input, int output) {
    output = input;
}

print (y);
fork write(var, y);
join;
print(y);
```

Generated SprIL

```
Compute Equal 1 0 6
Branch 6 (Rel 2)
Jump (Rel 7)
TestAndSet (DirAddr 2)
```

```
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 0) 7
Jump (Rel 111)
ReadInstr (DirAddr 0)
Receive 3
Compute Equal 3 0 6
Branch 6 (Rel 2)
EndProg
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-8))
ComputeI Add 1 30 3
TestAndSet (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
ReadInstr (DirAddr 3)
Receive 3
Push 3
ComputeI Add 7 1 4
ReadInstr (DirAddr 4)
Receive 5
Load (ImmValue 5) 2
Compute Equal 5 0 6
Branch 6 (Rel 18)
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
Compute Decr 5 0 5
Jump (Rel (-18))
Load (ImmValue 57) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
```

```
52 Store 7 (IndAddr 4)
53 Compute Add 4 0 7
54 Pop 2
55 WriteInstr 0 (DirAddr 1)
56 Jump (Ind 2)
57 ComputeI Add 1 30 3
58 WriteInstr 0 (IndAddr 3)
59 Jump (Abs 9)
60 Load (ImmValue 7) 2
61 Compute Sub 7 2 2
62 Load (ImmValue 1) 5
63 ComputeI Gt 5 2 6
64 Branch 6 (Rel 7)
65 Load (IndAddr 2) 3
66 Compute Add 7 5 6
67 Store 3 (IndAddr 6)
68 Compute Incr 5 0 5
69 ComputeI Add 2 3 2
70 Jump (Rel (-7))
71 Compute Add 7 0 4
72 ComputeI Add 4 3 4
73 Store 7 (IndAddr 4)
74 Compute Add 4 0 7
75 Compute Add 7 0 6
76 Load (IndAddr 6) 6
77 ComputeI Add 6 1 6
78 Load (IndAddr 6) 5
79 Push 5
80 Compute Add 7 0 6
81 Load (IndAddr 6) 6
82 ComputeI Add 6 2 6
83 Pop 2
84 Store 2 (IndAddr 6)
85 Push 2
86 Pop 0
87 Load (IndAddr 7) 7
88 Load (ImmValue 6) 2
89 Compute Sub 7 2 2
90 ComputeI Add 0 1 5
91 ComputeI Gt 5 2 6
92 Branch 6 (Rel 23)
93 Compute Add 7 5 6
94 Load (IndAddr 6) 4
95 Load (IndAddr 2) 3
96 Compute Lt 3 0 6
97 Branch 6 (Rel 2)
98 Store 4 (IndAddr 3)
99 Compute Incr 2 0 2
```

```
100 Load (IndAddr 2) 3
101 Compute Lt 3 0 6
102 Branch 6 (Rel 10)
103 Compute Add 3 0 6
104 TestAndSet (IndAddr 6)
105 Receive 6
106 Branch 6 (Rel 2)
107 Jump (Rel (-4))
108 ComputeI Add 3 1 3
109 WriteInstr 4 (IndAddr 3)
110 ComputeI Sub 3 1 3
111 WriteInstr 0 (IndAddr 3)
112 Compute Incr 5 0 5
113 ComputeI Add 2 2 2
114 Jump (Rel (-23))
115 Compute Decr 7 0 2
116 Load (IndAddr 2) 6
117 Load (IndAddr 7) 7
118 Jump (Ind 6)
119 Load (ImmValue 1337) 6
120 Push 6
121 Pop 6
122 Load (ImmValue 32) 2
123 TestAndSet (IndAddr 2)
124 Receive 3
125 Branch 3 (Rel 2)
126 Jump (Rel (-3))
127 Load (ImmValue 33) 4
128 WriteInstr 6 (IndAddr 4)
129 WriteInstr 0 (IndAddr 2)
130 Load (ImmValue 42) 6
131 Push 6
132 Pop 6
133 Load (ImmValue 34) 2
134 TestAndSet (IndAddr 2)
135 Receive 3
136 Branch 3 (Rel 2)
137 Jump (Rel (-3))
138 Load (ImmValue 35) 4
139 WriteInstr 6 (IndAddr 4)
140 WriteInstr 0 (IndAddr 2)
141 Load (ImmValue 34) 2
142 TestAndSet (IndAddr 2)
143 Receive 3
144 Branch 3 (Rel 2)
145 Jump (Rel (-4))
146 Load (ImmValue 35) 4
147 ReadInstr (IndAddr 4)
```

```
148 Receive 5
149 Push 5
150 WriteInstr 0 (IndAddr 2)
151 Pop 6
152 PrintOut 6
153 TestAndSet (DirAddr 1)
154 Receive 6
155 Branch 6 (Rel 2)
156 Jump (Rel (-3))
157 Load (ImmValue 34) 2
158 TestAndSet (IndAddr 2)
159 Receive 3
160 Branch 3 (Rel 2)
161 Jump (Rel (-4))
162 Load (ImmValue 35) 4
163 ReadInstr (IndAddr 4)
164 Receive 5
165 Push 5
166 WriteInstr 0 (IndAddr 2)
167 Load (ImmValue 32) 2
168 TestAndSet (IndAddr 2)
169 Receive 3
170 Branch 3 (Rel 2)
171 Jump (Rel (-4))
172 Load (ImmValue 33) 4
173 ReadInstr (IndAddr 4)
174 Receive 5
175 Push 5
176 WriteInstr 0 (IndAddr 2)
177 Load (ImmValue 5) 4
178 Pop 3
179 WriteInstr 3 (IndAddr 4)
180 Compute Incr 4 0 4
181 Load (ImmValue (-1)) 3
182 WriteInstr 3 (IndAddr 4)
183 Compute Incr 4 0 4
184 Load (ImmValue 32) 3
185 WriteInstr 3 (IndAddr 4)
186 Compute Incr 4 0 4
187 Pop 3
188 WriteInstr 3 (IndAddr 4)
189 Compute Incr 4 0 4
190 Load (ImmValue (-1)) 3
191 WriteInstr 3 (IndAddr 4)
192 Compute Incr 4 0 4
193 Load (ImmValue 34) 3
194 WriteInstr 3 (IndAddr 4)
195 Compute Incr 4 0 4
```

```
Load (ImmValue 2) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 60) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Compute Equal 0 1 6
Branch 6 (Rel 4)
Load (ImmValue 2) 2
PrintOut 2
EndProg
Load (ImmValue 30) 3
Load (ImmValue 0) 2
ReadInstr (IndAddr 3)
Receive 4
Compute Add 2 4 2
ComputeI NEq 3 32 6
Compute Incr 3 0 3
Branch 6 (Rel (-5))
Compute Equal 2 0 6
Branch 6 (Rel 2)
Jump (Rel (-10))
Load (ImmValue 34) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 35) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Pop 6
PrintOut 6
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 42
>>> 1337
```

Cyclic Recursion

Source

```
procedure prod(int i) {
    i = (i + 1);
    //print (i);
    cons(i);
}

procedure cons(int i) {
    if ((i > 1)) {
        i = (i - 2);
        print(i);
        prod(i);
    } else if ((i > 0)) {
        i = (i - 1);
        print(i);
    }
}

prod(18);
```

Generated SprIL

```
Branch 1 (Rel 6)
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Jump (Rel 323)
ReadInstr (DirAddr 0)
Receive 3
Compute Equal 3 0 6
Branch 6 (Rel 2)
EndProg
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-8))
ComputeI Add 1 30 3
TestAndSet (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
ReadInstr (DirAddr 3)
Receive 3
Push 3
```

```
ComputeI Add 7 1 4
ReadInstr (DirAddr 4)
Receive 5
Load (ImmValue 5) 2
Compute Equal 5 0 6
Branch 6 (Rel 18)
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
Compute Decr 5 0 5
Jump (Rel (-18))
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Load (ImmValue 4) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
ComputeI Gt 5 1 6
Branch 6 (Rel 7)
Load (IndAddr 2) 3
Compute Add 7 5 6
Store 3 (IndAddr 6)
Compute Incr 5 0 5
ComputeI Add 2 3 2
Jump (Rel (-7))
Compute Add 7 0 4
ComputeI Add 4 2 4
Store 7 (IndAddr 4)
```

```
Compute Add 4 0 7
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Load (ImmValue 1) 6
Push 6
Pop 3
Pop 2
Compute Add 2 3 4
Push 4
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Pop 2
Store 2 (IndAddr 6)
Push 2
Pop 0
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 4
ComputeI Add 4 1 4
Load (ImmValue 1) 5
Pop 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Store 6 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 120) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 152) 6
Push 6
Pop 2
```

```
119 Jump (Ind 2)
120 Load (IndAddr 7) 7
121 Load (ImmValue 3) 2
122 Compute Sub 7 2 2
123 ComputeI Add 0 1 5
124 ComputeI Gt 5 1 6
125 Branch 6 (Rel 23)
126 Compute Add 7 5 6
127 Load (IndAddr 6) 4
128 Load (IndAddr 2) 3
129 Compute Lt 3 0 6
130 Branch 6 (Rel 2)
131 Store 4 (IndAddr 3)
132 Compute Incr 2 0 2
133 Load (IndAddr 2) 3
134 Compute Lt 3 0 6
135 Branch 6 (Rel 10)
136 Compute Add 3 0 6
137 TestAndSet (IndAddr 6)
138 Receive 6
139 Branch 6 (Rel 2)
140 Jump (Rel (-4))
141 ComputeI Add 3 1 3
142 WriteInstr 4 (IndAddr 3)
143 ComputeI Sub 3 1 3
144 WriteInstr 0 (IndAddr 3)
145 Compute Incr 5 0 5
146 ComputeI Add 2 2 2
147 Jump (Rel (-23))
148 Compute Decr 7 0 2
149 Load (IndAddr 2) 6
150 Load (IndAddr 7) 7
151 Jump (Ind 6)
152 Load (ImmValue 4) 2
153 Compute Sub 7 2 2
154 Load (ImmValue 1) 5
155 ComputeI Gt 5 1 6
156 Branch 6 (Rel 7)
157 Load (IndAddr 2) 3
158 Compute Add 7 5 6
159 Store 3 (IndAddr 6)
160 Compute Incr 5 0 5
161 ComputeI Add 2 3 2
162 Jump (Rel (-7))
163 Compute Add 7 0 4
164 ComputeI Add 4 2 4
165 Store 7 (IndAddr 4)
166 Compute Add 4 0 7
```

```
167 Compute Add 7 0 6
168 Load (IndAddr 6) 6
169 ComputeI Add 6 1 6
170 Load (IndAddr 6) 5
171 Push 5
172 Load (ImmValue 1) 6
173 Push 6
174 Pop 3
175 Pop 2
176 Compute Gt 2 3 4
177 Push 4
178 Pop 6
179 ComputeI Xor 6 1 6
180 Branch 6 (Rel 67)
181 Compute Add 7 0 4
182 ComputeI Add 4 1 4
183 Store 7 (IndAddr 4)
184 Compute Add 4 0 7
185 Compute Add 7 0 6
186 Load (IndAddr 6) 6
187 Load (IndAddr 6) 6
188 ComputeI Add 6 1 6
189 Load (IndAddr 6) 5
190 Push 5
191 Load (ImmValue 2) 6
192 Push 6
193 Pop 3
194 Pop 2
195 Compute Sub 2 3 4
196 Push 4
197 Compute Add 7 0 6
198 Load (IndAddr 6) 6
199 Load (IndAddr 6) 6
200 ComputeI Add 6 1 6
201 Pop 2
202 Store 2 (IndAddr 6)
203 Push 2
204 Pop 0
205 Compute Add 7 0 6
206 Load (IndAddr 6) 6
207 Load (IndAddr 6) 6
208 ComputeI Add 6 1 6
209 Load (IndAddr 6) 5
210 Push 5
211 Pop 6
212 PrintOut 6
213 Compute Add 7 0 6
214 Load (IndAddr 6) 6
```

```
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 4
ComputeI Add 4 1 4
Load (ImmValue 1) 5
Pop 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Store 6 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 245) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 57) 6
Push 6
Pop 2
Jump (Ind 2)
Load (IndAddr 7) 7
Jump (Rel 48)
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Load (ImmValue 0) 6
Push 6
Pop 3
Pop 2
Compute Gt 2 3 4
Push 4
Pop 6
ComputeI Xor 6 1 6
Branch 6 (Rel 34)
Compute Add 7 0 4
ComputeI Add 4 1 4
```

```
Store 7 (IndAddr 4)
Compute Add 4 0 7
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Load (ImmValue 1) 6
Push 6
Pop 3
Pop 2
Compute Sub 2 3 4
Push 4
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Pop 2
Store 2 (IndAddr 6)
Push 2
Pop 0
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
PrintOut 6
Load (IndAddr 7) 7
Load (IndAddr 7) 7
Load (ImmValue 3) 2
Compute Sub 7 2 2
ComputeI Add 0 1 5
ComputeI Gt 5 1 6
Branch 6 (Rel 23)
Compute Add 7 5 6
Load (IndAddr 6) 4
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 2)
Store 4 (IndAddr 3)
Compute Incr 2 0 2
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 10)
Compute Add 3 0 6
```

```
311 TestAndSet (IndAddr 6)
312 Receive 6
313 Branch 6 (Rel 2)
314 Jump (Rel (-4))
315 ComputeI Add 3 1 3
316 WriteInstr 4 (IndAddr 3)
317 ComputeI Sub 3 1 3
318 WriteInstr 0 (IndAddr 3)
319 Compute Incr 5 0 5
320 ComputeI Add 2 2 2
321 Jump (Rel (-23))
322 Compute Decr 7 0 2
323 Load (IndAddr 2) 6
324 Load (IndAddr 7) 7
325 Jump (Ind 6)
326 Nop
327 Nop
328 Load (ImmValue 18) 6
329 Push 6
330 Compute Add 7 0 4
331 ComputeI Add 4 1 4
332 Load (ImmValue 1) 5
333 Pop 3
334 Store 3 (IndAddr 4)
335 Compute Incr 4 0 4
336 Load (ImmValue (-1)) 3
337 Store 3 (IndAddr 4)
338 Compute Incr 4 0 4
339 Load (ImmValue (-1)) 3
340 Store 3 (IndAddr 4)
341 Compute Incr 4 0 4
342 Load (ImmValue 353) 6
343 Push 6
344 Pop 5
345 Store 5 (IndAddr 4)
346 Compute Incr 4 0 4
347 Store 7 (IndAddr 4)
348 Compute Add 4 0 7
349 Load (ImmValue 57) 6
350 Push 6
351 Pop 2
352 Jump (Ind 2)
353 Load (ImmValue 1) 2
354 WriteInstr 2 (DirAddr 0)
355 EndProg
```

Results

```
>>> 17
>>> 16
>>> 15
>>> 14
>>> 13
>>> 12
>>> 11
>>> 10
>>> 9
>>> 8
>>> 7
>>> 6
>>> 5
>>> 4
>>> 3
>>> 2
>>> 1
>>> 0
>>> 0
```

Deep Expression

Source

```
int a = 100;
100000;
200000;
300000;
400000;
a = ((a + (((10 * (-15)) * 42) * (3 + 2))) * (2 * ((7 + 11) - 98))); // actual approximate answer fo
print(a);
```

Generated SprIL

```
Branch 1 (Rel 6)
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Jump (Rel 54)
ReadInstr (DirAddr 0)
Receive 3
Compute Equal 3 0 6
Branch 6 (Rel 2)
EndProg
TestAndSet (DirAddr 2)
```

```
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-8))
ComputeI Add 1 30 3
TestAndSet (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
ReadInstr (DirAddr 3)
Receive 3
Push 3
ComputeI Add 7 1 4
ReadInstr (DirAddr 4)
Receive 5
Load (ImmValue 5) 2
Compute Equal 5 0 6
Branch 6 (Rel 18)
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
Compute Decr 5 0 5
Jump (Rel (-18))
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Nop
Nop
Load (ImmValue 100) 6
```

```
Push 6
Compute Add 7 0 6
ComputeI Add 6 1 6
Pop 5
Store 5 (IndAddr 6)
Load (ImmValue 100000) 6
Push 6
Pop 0
Load (ImmValue 200000) 6
Push 6
Pop 0
Load (ImmValue 300000) 6
Push 6
Pop 0
Load (ImmValue 400000) 6
Push 6
Pop 0
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Load (ImmValue 10) 6
Push 6
Load (ImmValue 15) 6
Push 6
Pop 2
Compute Sub 2 2 4
Compute Sub 4 2 4
Push 4
Pop 3
Pop 2
Compute Mul 2 3 4
Push 4
Load (ImmValue 42) 6
Push 6
Pop 3
Pop 2
Compute Mul 2 3 4
Push 4
Load (ImmValue 3) 6
Push 6
Load (ImmValue 2) 6
Push 6
Pop 3
Pop 2
Compute Add 2 3 4
Push 4
Pop 3
```

```
Pop 2
Compute Mul 2 3 4
Push 4
Pop 3
Pop 2
Compute Add 2 3 4
Push 4
Load (ImmValue 2) 6
Push 6
Load (ImmValue 7) 6
Push 6
Load (ImmValue 11) 6
Push 6
Pop 3
Pop 2
Compute Add 2 3 4
Push 4
Load (ImmValue 98) 6
Push 6
Pop 3
Pop 2
Compute Sub 2 3 4
Push 4
Pop 3
Pop 2
Compute Mul 2 3 4
Push 4
Pop 3
Pop 2
Compute Mul 2 3 4
Push 4
Compute Add 7 0 6
ComputeI Add 6 1 6
Pop 2
Store 2 (IndAddr 6)
Push 2
Pop 0
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
PrintOut 6
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 5024000
```

Fib

Source

```
procedure fib(int i, int res) {
    if ((i < (3))) {
        res = 1;
    } else {
        int a;
        int b;
        fib((i-1), a);
        fib((i-2), b);
        res = (a + b);

    }
}

int a = 0;
fib(8, a);
print(a);
```

Generated SprIL

```
Branch 1 (Rel 6)
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Jump (Rel 260)
ReadInstr (DirAddr 0)
Receive 3
Compute Equal 3 0 6
Branch 6 (Rel 2)
EndProg
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-8))
ComputeI Add 1 30 3
TestAndSet (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
ReadInstr (DirAddr 3)
```

```
21 Receive 3
22 Push 3
23 ComputeI Add 7 1 4
24 ReadInstr (DirAddr 4)
25 Receive 5
26 Load (ImmValue 5) 2
27 Compute Equal 5 0 6
28 Branch 6 (Rel 18)
29 ReadInstr (IndAddr 2)
30 Receive 3
31 Store 3 (IndAddr 4)
32 Compute Incr 2 0 2
33 Compute Incr 4 0 4
34 ReadInstr (IndAddr 2)
35 Receive 3
36 Store 3 (IndAddr 4)
37 Compute Incr 2 0 2
38 Compute Incr 4 0 4
39 ReadInstr (IndAddr 2)
40 Receive 3
41 Store 3 (IndAddr 4)
42 Compute Incr 2 0 2
43 Compute Incr 4 0 4
44 Compute Decr 5 0 5
45 Jump (Rel (-18))
46 Load (ImmValue 54) 5
47 Store 5 (IndAddr 4)
48 Compute Incr 4 0 4
49 Store 7 (IndAddr 4)
50 Compute Add 4 0 7
51 Pop 2
52 WriteInstr 0 (DirAddr 1)
53 Jump (Ind 2)
54 ComputeI Add 1 30 3
55 WriteInstr 0 (IndAddr 3)
56 Jump (Abs 9)
57 Load (ImmValue 7) 2
58 Compute Sub 7 2 2
59 Load (ImmValue 1) 5
60 ComputeI Gt 5 2 6
61 Branch 6 (Rel 7)
62 Load (IndAddr 2) 3
63 Compute Add 7 5 6
64 Store 3 (IndAddr 6)
65 Compute Incr 5 0 5
66 ComputeI Add 2 3 2
67 Jump (Rel (-7))
68 Compute Add 7 0 4
```

```
69  ComputeI Add 4 3 4
70  Store 7 (IndAddr 4)
71  Compute Add 4 0 7
72  Compute Add 7 0 6
73  Load (IndAddr 6) 6
74  ComputeI Add 6 1 6
75  Load (IndAddr 6) 5
76  Push 5
77  Load (ImmValue 3) 6
78  Push 6
79  Pop 3
80  Pop 2
81  Compute Lt 2 3 4
82  Push 4
83  Pop 6
84  ComputeI Xor 6 1 6
85  Branch 6 (Rel 17)
86  Compute Add 7 0 4
87  ComputeI Add 4 1 4
88  Store 7 (IndAddr 4)
89  Compute Add 4 0 7
90  Load (ImmValue 1) 6
91  Push 6
92  Compute Add 7 0 6
93  Load (IndAddr 6) 6
94  Load (IndAddr 6) 6
95  ComputeI Add 6 2 6
96  Pop 2
97  Store 2 (IndAddr 6)
98  Push 2
99  Pop 0
100 Load (IndAddr 7) 7
101 Jump (Rel 130)
102 Compute Add 7 0 4
103 ComputeI Add 4 1 4
104 Store 7 (IndAddr 4)
105 Compute Add 4 0 7
106 Compute Add 7 0 6
107 ComputeI Add 6 1 6
108 Store 0 (IndAddr 6)
109 Compute Add 7 0 6
110 ComputeI Add 6 2 6
111 Store 0 (IndAddr 6)
112 Compute Add 7 0 6
113 ComputeI Add 6 1 6
114 Load (IndAddr 6) 5
115 Push 5
116 Compute Add 7 0 6
```

```
117 Load (IndAddr 6) 6
118 Load (IndAddr 6) 6
119 ComputeI Add 6 1 6
120 Load (IndAddr 6) 5
121 Push 5
122 Load (ImmValue 1) 6
123 Push 6
124 Pop 3
125 Pop 2
126 Compute Sub 2 3 4
127 Push 4
128 Compute Add 7 0 4
129 ComputeI Add 4 3 4
130 Load (ImmValue 2) 5
131 Pop 3
132 Store 3 (IndAddr 4)
133 Compute Incr 4 0 4
134 Load (ImmValue (-1)) 3
135 Store 3 (IndAddr 4)
136 Compute Incr 4 0 4
137 Load (ImmValue (-1)) 3
138 Store 3 (IndAddr 4)
139 Compute Incr 4 0 4
140 Pop 3
141 Store 3 (IndAddr 4)
142 Compute Incr 4 0 4
143 Compute Add 7 0 6
144 ComputeI Add 6 1 6
145 Store 6 (IndAddr 4)
146 Compute Incr 4 0 4
147 Load (ImmValue (-1)) 3
148 Store 3 (IndAddr 4)
149 Compute Incr 4 0 4
150 Load (ImmValue 161) 6
151 Push 6
152 Pop 5
153 Store 5 (IndAddr 4)
154 Compute Incr 4 0 4
155 Store 7 (IndAddr 4)
156 Compute Add 4 0 7
157 Load (ImmValue 57) 6
158 Push 6
159 Pop 2
160 Jump (Ind 2)
161 Compute Add 7 0 6
162 ComputeI Add 6 2 6
163 Load (IndAddr 6) 5
164 Push 5
```

```
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Load (ImmValue 2) 6
Push 6
Pop 3
Pop 2
Compute Sub 2 3 4
Push 4
Compute Add 7 0 4
ComputeI Add 4 3 4
Load (ImmValue 2) 5
Pop 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Compute Add 7 0 6
ComputeI Add 6 2 6
Store 6 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 210) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 57) 6
Push 6
Pop 2
Jump (Ind 2)
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
```

```
213 Push 5
214 Compute Add 7 0 6
215 ComputeI Add 6 2 6
216 Load (IndAddr 6) 5
217 Push 5
218 Pop 3
219 Pop 2
220 Compute Add 2 3 4
221 Push 4
222 Compute Add 7 0 6
223 Load (IndAddr 6) 6
224 Load (IndAddr 6) 6
225 ComputeI Add 6 2 6
226 Pop 2
227 Store 2 (IndAddr 6)
228 Push 2
229 Pop 0
230 Load (IndAddr 7) 7
231 Load (IndAddr 7) 7
232 Load (ImmValue 6) 2
233 Compute Sub 7 2 2
234 ComputeI Add 0 1 5
235 ComputeI Gt 5 2 6
236 Branch 6 (Rel 23)
237 Compute Add 7 5 6
238 Load (IndAddr 6) 4
239 Load (IndAddr 2) 3
240 Compute Lt 3 0 6
241 Branch 6 (Rel 2)
242 Store 4 (IndAddr 3)
243 Compute Incr 2 0 2
244 Load (IndAddr 2) 3
245 Compute Lt 3 0 6
246 Branch 6 (Rel 10)
247 Compute Add 3 0 6
248 TestAndSet (IndAddr 6)
249 Receive 6
250 Branch 6 (Rel 2)
251 Jump (Rel (-4))
252 ComputeI Add 3 1 3
253 WriteInstr 4 (IndAddr 3)
254 ComputeI Sub 3 1 3
255 WriteInstr 0 (IndAddr 3)
256 Compute Incr 5 0 5
257 ComputeI Add 2 2 2
258 Jump (Rel (-23))
259 Compute Decr 7 0 2
260 Load (IndAddr 2) 6
```

```
Load (IndAddr 7) 7
Jump (Ind 6)
Nop
Nop
Load (ImmValue 0) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 1 6
Pop 5
Store 5 (IndAddr 6)
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Load (ImmValue 8) 6
Push 6
Compute Add 7 0 4
ComputeI Add 4 2 4
Load (ImmValue 2) 5
Pop 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Pop 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Compute Add 7 0 6
ComputeI Add 6 1 6
Store 6 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 310) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 57) 6
Push 6
Pop 2
```

```
Jump (Ind 2)
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
PrintOut 6
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 21
```

If

Source

```
bool condition = true;
if (condition) print(1); else print(0);
```

Generated SprIL

```
Branch 1 (Rel 6)
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Jump (Rel 54)
ReadInstr (DirAddr 0)
Receive 3
Compute Equal 3 0 6
Branch 6 (Rel 2)
EndProg
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-8))
ComputeI Add 1 30 3
TestAndSet (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
ReadInstr (DirAddr 3)
Receive 3
Push 3
ComputeI Add 7 1 4
```

```
ReadInstr (DirAddr 4)
Receive 5
Load (ImmValue 5) 2
Compute Equal 5 0 6
Branch 6 (Rel 18)
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
Compute Decr 5 0 5
Jump (Rel (-18))
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Nop
Nop
Load (ImmValue 1) 6
Push 6
Compute Add 7 0 6
ComputeI Add 6 1 6
Pop 5
Store 5 (IndAddr 6)
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
ComputeI Xor 6 1 6
Branch 6 (Rel 6)
```

```
Load (ImmValue 1) 6
Push 6
Pop 6
PrintOut 6
Jump (Rel 5)
Load (ImmValue 0) 6
Push 6
Pop 6
PrintOut 6
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 1
```

If Else

Source

```
int i = 4;
print (i);
if ((i == 2)) {
    print(3,i);
} else if ((i == 1)) {{{
    print(4,i);
}}} else {{{
    print(5,i);
}}}


if ((i == 4)) {
    print(3,i);
} else if ((i == 1)) {{{
    print(4,i);
}}}
```

Generated SprIL

```
Branch 1 (Rel 6)
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Jump (Rel 54)
ReadInstr (DirAddr 0)
Receive 3
```

```
Compute Equal 3 0 6
Branch 6 (Rel 2)
EndProg
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-8))
ComputeI Add 1 30 3
TestAndSet (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
ReadInstr (DirAddr 3)
Receive 3
Push 3
ComputeI Add 7 1 4
ReadInstr (DirAddr 4)
Receive 5
Load (ImmValue 5) 2
Compute Equal 5 0 6
Branch 6 (Rel 18)
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
Compute Decr 5 0 5
Jump (Rel (-18))
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
```

```
56  Jump (Abs 9)
57  Nop
58  Nop
59  Load (ImmValue 4) 6
60  Push 6
61  Compute Add 7 0 6
62  ComputeI Add 6 1 6
63  Pop 5
64  Store 5 (IndAddr 6)
65  Compute Add 7 0 6
66  ComputeI Add 6 1 6
67  Load (IndAddr 6) 5
68  Push 5
69  Pop 6
70  PrintOut 6
71  Compute Add 7 0 6
72  ComputeI Add 6 1 6
73  Load (IndAddr 6) 5
74  Push 5
75  Load (ImmValue 2) 6
76  Push 6
77  Pop 3
78  Pop 2
79  Compute Equal 2 3 4
80  Push 4
81  Pop 6
82  ComputeI Xor 6 1 6
83  Branch 6 (Rel 18)
84  Compute Add 7 0 4
85  ComputeI Add 4 2 4
86  Store 7 (IndAddr 4)
87  Compute Add 4 0 7
88  Compute Add 7 0 6
89  Load (IndAddr 6) 6
90  ComputeI Add 6 1 6
91  Load (IndAddr 6) 5
92  Push 5
93  Load (ImmValue 3) 6
94  Push 6
95  Pop 6
96  PrintOut 6
97  Pop 6
98  PrintOut 6
99  Load (IndAddr 7) 7
100 Jump (Rel 71)
101 Compute Add 7 0 6
102 ComputeI Add 6 1 6
103 Load (IndAddr 6) 5
```

```
104 Push 5
105 Load (ImmValue 1) 6
106 Push 6
107 Pop 3
108 Pop 2
109 Compute Equal 2 3 4
110 Push 4
111 Pop 6
112 ComputeI Xor 6 1 6
113 Branch 6 (Rel 30)
114 Compute Add 7 0 4
115 ComputeI Add 4 2 4
116 Store 7 (IndAddr 4)
117 Compute Add 4 0 7
118 Compute Add 7 0 4
119 ComputeI Add 4 1 4
120 Store 7 (IndAddr 4)
121 Compute Add 4 0 7
122 Compute Add 7 0 4
123 ComputeI Add 4 1 4
124 Store 7 (IndAddr 4)
125 Compute Add 4 0 7
126 Compute Add 7 0 6
127 Load (IndAddr 6) 6
128 Load (IndAddr 6) 6
129 Load (IndAddr 6) 6
130 ComputeI Add 6 1 6
131 Load (IndAddr 6) 5
132 Push 5
133 Load (ImmValue 4) 6
134 Push 6
135 Pop 6
136 PrintOut 6
137 Pop 6
138 PrintOut 6
139 Load (IndAddr 7) 7
140 Load (IndAddr 7) 7
141 Load (IndAddr 7) 7
142 Jump (Rel 29)
143 Compute Add 7 0 4
144 ComputeI Add 4 2 4
145 Store 7 (IndAddr 4)
146 Compute Add 4 0 7
147 Compute Add 7 0 4
148 ComputeI Add 4 1 4
149 Store 7 (IndAddr 4)
150 Compute Add 4 0 7
151 Compute Add 7 0 4
```

```
152 ComputeI Add 4 1 4
153 Store 7 (IndAddr 4)
154 Compute Add 4 0 7
155 Compute Add 7 0 6
156 Load (IndAddr 6) 6
157 Load (IndAddr 6) 6
158 Load (IndAddr 6) 6
159 ComputeI Add 6 1 6
160 Load (IndAddr 6) 5
161 Push 5
162 Load (ImmValue 5) 6
163 Push 6
164 Pop 6
165 PrintOut 6
166 Pop 6
167 PrintOut 6
168 Load (IndAddr 7) 7
169 Load (IndAddr 7) 7
170 Load (IndAddr 7) 7
171 Compute Add 7 0 6
172 ComputeI Add 6 1 6
173 Load (IndAddr 6) 5
174 Push 5
175 Load (ImmValue 4) 6
176 Push 6
177 Pop 3
178 Pop 2
179 Compute Equal 2 3 4
180 Push 4
181 Pop 6
182 ComputeI Xor 6 1 6
183 Branch 6 (Rel 18)
184 Compute Add 7 0 4
185 ComputeI Add 4 2 4
186 Store 7 (IndAddr 4)
187 Compute Add 4 0 7
188 Compute Add 7 0 6
189 Load (IndAddr 6) 6
190 ComputeI Add 6 1 6
191 Load (IndAddr 6) 5
192 Push 5
193 Load (ImmValue 3) 6
194 Push 6
195 Pop 6
196 PrintOut 6
197 Pop 6
198 PrintOut 6
199 Load (IndAddr 7) 7
```

```
Jump (Rel 42)
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Load (ImmValue 1) 6
Push 6
Pop 3
Pop 2
Compute Equal 2 3 4
Push 4
Pop 6
ComputeI Xor 6 1 6
Branch 6 (Rel 29)
Compute Add 7 0 4
ComputeI Add 4 2 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Compute Add 7 0 4
ComputeI Add 4 1 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Compute Add 7 0 4
ComputeI Add 4 1 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Load (ImmValue 4) 6
Push 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Load (IndAddr 7) 7
Load (IndAddr 7) 7
Load (IndAddr 7) 7
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 4
>>> 5
>>> 4
>>> 3
>>> 4
```

Infinite Busy Loop

Source

```
int i = 0;
int j = 1;
while (true) {
    i = (i + j);
    j = (j * i);
    print(i,j);
}
```

Generated SprIL

```
Branch 1 (Rel 6)
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Jump (Rel 54)
ReadInstr (DirAddr 0)
Receive 3
Compute Equal 3 0 6
Branch 6 (Rel 2)
EndProg
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-8))
ComputeI Add 1 30 3
TestAndSet (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
ReadInstr (DirAddr 3)
Receive 3
Push 3
ComputeI Add 7 1 4
ReadInstr (DirAddr 4)
Receive 5
```

```
26 Load (ImmValue 5) 2
27 Compute Equal 5 0 6
28 Branch 6 (Rel 18)
29 ReadInstr (IndAddr 2)
30 Receive 3
31 Store 3 (IndAddr 4)
32 Compute Incr 2 0 2
33 Compute Incr 4 0 4
34 ReadInstr (IndAddr 2)
35 Receive 3
36 Store 3 (IndAddr 4)
37 Compute Incr 2 0 2
38 Compute Incr 4 0 4
39 ReadInstr (IndAddr 2)
40 Receive 3
41 Store 3 (IndAddr 4)
42 Compute Incr 2 0 2
43 Compute Incr 4 0 4
44 Compute Decr 5 0 5
45 Jump (Rel (-18))
46 Load (ImmValue 54) 5
47 Store 5 (IndAddr 4)
48 Compute Incr 4 0 4
49 Store 7 (IndAddr 4)
50 Compute Add 4 0 7
51 Pop 2
52 WriteInstr 0 (DirAddr 1)
53 Jump (Ind 2)
54 ComputeI Add 1 30 3
55 WriteInstr 0 (IndAddr 3)
56 Jump (Abs 9)
57 Nop
58 Nop
59 Load (ImmValue 0) 6
60 Push 6
61 Compute Add 7 0 6
62 ComputeI Add 6 1 6
63 Pop 5
64 Store 5 (IndAddr 6)
65 Load (ImmValue 1) 6
66 Push 6
67 Compute Add 7 0 6
68 ComputeI Add 6 2 6
69 Pop 5
70 Store 5 (IndAddr 6)
71 Load (ImmValue 1) 6
72 Push 6
73 Pop 6
```

```
74  ComputeI Xor 6 1 6
75  Branch 6 (Rel 63)
76  Compute Add 7 0 4
77  ComputeI Add 4 3 4
78  Store 7 (IndAddr 4)
79  Compute Add 4 0 7
80  Compute Add 7 0 6
81  Load (IndAddr 6) 6
82  ComputeI Add 6 1 6
83  Load (IndAddr 6) 5
84  Push 5
85  Compute Add 7 0 6
86  Load (IndAddr 6) 6
87  ComputeI Add 6 2 6
88  Load (IndAddr 6) 5
89  Push 5
90  Pop 3
91  Pop 2
92  Compute Add 2 3 4
93  Push 4
94  Compute Add 7 0 6
95  Load (IndAddr 6) 6
96  ComputeI Add 6 1 6
97  Pop 2
98  Store 2 (IndAddr 6)
99  Push 2
100 Pop 0
101 Compute Add 7 0 6
102 Load (IndAddr 6) 6
103 ComputeI Add 6 2 6
104 Load (IndAddr 6) 5
105 Push 5
106 Compute Add 7 0 6
107 Load (IndAddr 6) 6
108 ComputeI Add 6 1 6
109 Load (IndAddr 6) 5
110 Push 5
111 Pop 3
112 Pop 2
113 Compute Mul 2 3 4
114 Push 4
115 Compute Add 7 0 6
116 Load (IndAddr 6) 6
117 ComputeI Add 6 2 6
118 Pop 2
119 Store 2 (IndAddr 6)
120 Push 2
121 Pop 0
```

```
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 2 6
Load (IndAddr 6) 5
Push 5
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Load (IndAddr 7) 7
Jump (Rel (-66))
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

Gets stuck in an infinite loop, repeating the same output.

```
>>> 1
>>> 1
>>> 2
>>> 2
>>> 4
>>> 8
>>> 12
>>> 96
>>> 108
>>> 10368
>>> 10476
>>> 108615168
>>> 108625644
>>> 11798392572168192
>>> 11798392680793836
>>> -5570361874949185536
>>> -5558563482268391700
>>> 3671369242980155392
>>> -1887194239288236308
>>> -4483044364780175360
>>> -6370238604068411668
>>> -8730959061097906176
>>> 3345546408543233772
>>> -6745737849034768384
>>> -3400191440491534612
```

```
>>> -6096120617457680384
>>> 8950432015760336620
>>> -1019520187243692032
>>> 7930911828516644588
>>> -4809903748681826304
>>> 3121008079834818284
>>> 5865085819223539712
>>> 8986093899058357996
>>> 2740241432517279744
>>> -6720408742133913876
>>> 3246081813541552128
>>> -3474326928592361748
>>> -1859074291971129344
>>> -5333401220563491092
>>> 681350175863603200
>>> -4652051044699887892
>>> -8143132099134619648
>>> 5651560929875044076
>>> 6951259845357993984
>>> -5843923298476513556
>>> 4700992750881865728
>>> -1142930547594647828
>>> -8561800288468467712
>>> 8742013237646436076
>>> 7566188111470788608
>>> -2138542724592326932
>>> -6956372574427152384
>>> -9094915299019479316
>>> -8878846665360932864
>>> 472982109329139436
>>> 1756403854674493440
>>> 2229385964003632876
>>> -5152117973711847424
>>> -2922732009708214548
>>> 1585267068834414592
>>> -1337464940873799956
>>> 5188146770730811392
>>> 3850681829857011436
>>> 6917529027641081856
>>> -7678533216211458324
>>> -9223372036854775808
>>> 1544838820643317484
>>> 0
>>> 1544838820643317484
>>> 0
>>> 1544838820643317484
>>> 0
>>> 1544838820643317484
```

```
>>> 0
\ldots
```

Infinite Empty Loop

Source

```
while (true) {
    // do nothing
}
```

Generated SprIL

```
Branch 1 (Rel 6)
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Jump (Rel 54)
ReadInstr (DirAddr 0)
Receive 3
Compute Equal 3 0 6
Branch 6 (Rel 2)
EndProg
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-8))
ComputeI Add 1 30 3
TestAndSet (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
ReadInstr (DirAddr 3)
Receive 3
Push 3
ComputeI Add 7 1 4
ReadInstr (DirAddr 4)
Receive 5
Load (ImmValue 5) 2
Compute Equal 5 0 6
Branch 6 (Rel 18)
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
```

```
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
Compute Decr 5 0 5
Jump (Rel (-18))
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Nop
Nop
Load (ImmValue 1) 6
Push 6
Pop 6
ComputeI Xor 6 1 6
Branch 6 (Rel 7)
Compute Add 7 0 4
ComputeI Add 4 1 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (IndAddr 7) 7
Jump (Rel (-10))
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

No output, gets stuck in an infinite loop.

Join Test

Source

```
1  global bool forever = true;
2
3  procedure neverending(bool j) {
4      while(j){
5
6      }
7  }
8
9  fork neverending(forever);
10 join;
11 print(1);
```

Generated SprIL

```
0  Branch 1 (Rel 6)
1  TestAndSet (DirAddr 2)
2  Receive 6
3  Branch 6 (Rel 2)
4  Jump (Rel (-3))
5  Jump (Rel 115)
6  ReadInstr (DirAddr 0)
7  Receive 3
8  Compute Equal 3 0 6
9  Branch 6 (Rel 2)
10 EndProg
11 TestAndSet (DirAddr 2)
12 Receive 6
13 Branch 6 (Rel 2)
14 Jump (Rel (-8))
15 ComputeI Add 1 30 3
16 TestAndSet (IndAddr 3)
17 Receive 6
18 Branch 6 (Rel 2)
19 Jump (Rel (-3))
20 ReadInstr (DirAddr 3)
21 Receive 3
22 Push 3
23 ComputeI Add 7 1 4
24 ReadInstr (DirAddr 4)
25 Receive 5
26 Load (ImmValue 5) 2
27 Compute Equal 5 0 6
28 Branch 6 (Rel 18)
29 ReadInstr (IndAddr 2)
```

```
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
Compute Decr 5 0 5
Jump (Rel (-18))
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Load (ImmValue 4) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
ComputeI Gt 5 1 6
Branch 6 (Rel 7)
Load (IndAddr 2) 3
Compute Add 7 5 6
Store 3 (IndAddr 6)
Compute Incr 5 0 5
ComputeI Add 2 3 2
Jump (Rel (-7))
Compute Add 7 0 4
ComputeI Add 4 2 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
```

```
78  ComputeI Xor 6 1 6
79  Branch 6 (Rel 7)
80  Compute Add 7 0 4
81  ComputeI Add 4 1 4
82  Store 7 (IndAddr 4)
83  Compute Add 4 0 7
84  Load (IndAddr 7) 7
85  Jump (Rel (-13))
86  Load (IndAddr 7) 7
87  Load (ImmValue 3) 2
88  Compute Sub 7 2 2
89  ComputeI Add 0 1 5
90  ComputeI Gt 5 1 6
91  Branch 6 (Rel 23)
92  Compute Add 7 5 6
93  Load (IndAddr 6) 4
94  Load (IndAddr 2) 3
95  Compute Lt 3 0 6
96  Branch 6 (Rel 2)
97  Store 4 (IndAddr 3)
98  Compute Incr 2 0 2
99  Load (IndAddr 2) 3
100 Compute Lt 3 0 6
101 Branch 6 (Rel 10)
102 Compute Add 3 0 6
103 TestAndSet (IndAddr 6)
104 Receive 6
105 Branch 6 (Rel 2)
106 Jump (Rel (-4))
107 ComputeI Add 3 1 3
108 WriteInstr 4 (IndAddr 3)
109 ComputeI Sub 3 1 3
110 WriteInstr 0 (IndAddr 3)
111 Compute Incr 5 0 5
112 ComputeI Add 2 2 2
113 Jump (Rel (-23))
114 Compute Decr 7 0 2
115 Load (IndAddr 2) 6
116 Load (IndAddr 7) 7
117 Jump (Ind 6)
118 Nop
119 Nop
120 Load (ImmValue 1) 6
121 Push 6
122 Pop 6
123 Load (ImmValue 32) 2
124 TestAndSet (IndAddr 2)
125 Receive 3
```

```
126 Branch 3 (Rel 2)
127 Jump (Rel (-3))
128 Load (ImmValue 33) 4
129 WriteInstr 6 (IndAddr 4)
130 WriteInstr 0 (IndAddr 2)
131 TestAndSet (DirAddr 1)
132 Receive 6
133 Branch 6 (Rel 2)
134 Jump (Rel (-3))
135 Load (ImmValue 32) 2
136 TestAndSet (IndAddr 2)
137 Receive 3
138 Branch 3 (Rel 2)
139 Jump (Rel (-4))
140 Load (ImmValue 33) 4
141 ReadInstr (IndAddr 4)
142 Receive 5
143 Push 5
144 WriteInstr 0 (IndAddr 2)
145 Load (ImmValue 5) 4
146 Pop 3
147 WriteInstr 3 (IndAddr 4)
148 Compute Incr 4 0 4
149 Load (ImmValue (-1)) 3
150 WriteInstr 3 (IndAddr 4)
151 Compute Incr 4 0 4
152 Load (ImmValue 32) 3
153 WriteInstr 3 (IndAddr 4)
154 Compute Incr 4 0 4
155 Load (ImmValue 1) 5
156 WriteInstr 5 (DirAddr 4)
157 Load (ImmValue 57) 6
158 Push 6
159 Pop 5
160 WriteInstr 5 (DirAddr 3)
161 WriteInstr 0 (DirAddr 2)
162 Load (ImmValue 1) 3
163 ReadInstr (IndAddr 3)
164 Receive 6
165 Branch 6 (Rel 2)
166 Jump (Rel (-3))
167 Compute Equal 0 1 6
168 Branch 6 (Rel 4)
169 Load (ImmValue 2) 2
170 PrintOut 2
171 EndProg
172 Load (ImmValue 30) 3
173 Load (ImmValue 0) 2
```

```
ReadInstr (IndAddr 3)
Receive 4
Compute Add 2 4 2
ComputeI NEq 3 32 6
Compute Incr 3 0 3
Branch 6 (Rel (-5))
Compute Equal 2 0 6
Branch 6 (Rel 2)
Jump (Rel (-10))
Load (ImmValue 1) 6
Push 6
Pop 6
PrintOut 6
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

No output, gets stuck in an infinite loop.

Multiple Globals

Source

```
global int a = 8;
global int b = 9;
global int c = 10;
global int d = 11;
global int e = 12;
global int f = 13;

procedure printAll() {
    print(a,b,c,d,e,f);
}

procedure printAllBW() {
    print(f,e,d,c,b,a);
}


fork printAll();
fork printAllBW();
join;
```

Generated SprIL

```
0  Branch 1 (Rel 6)
1  TestAndSet (DirAddr 2)
2  Receive 6
3  Branch 6 (Rel 2)
4  Jump (Rel (-3))
5  Jump (Rel 292)
6  ReadInstr (DirAddr 0)
7  Receive 3
8  Compute Equal 3 0 6
9  Branch 6 (Rel 2)
10 EndProg
11 TestAndSet (DirAddr 2)
12 Receive 6
13 Branch 6 (Rel 2)
14 Jump (Rel (-8))
15 ComputeI Add 1 30 3
16 TestAndSet (IndAddr 3)
17 Receive 6
18 Branch 6 (Rel 2)
19 Jump (Rel (-3))
20 ReadInstr (DirAddr 3)
21 Receive 3
22 Push 3
23 ComputeI Add 7 1 4
24 ReadInstr (DirAddr 4)
25 Receive 5
26 Load (ImmValue 5) 2
27 Compute Equal 5 0 6
28 Branch 6 (Rel 18)
29 ReadInstr (IndAddr 2)
30 Receive 3
31 Store 3 (IndAddr 4)
32 Compute Incr 2 0 2
33 Compute Incr 4 0 4
34 ReadInstr (IndAddr 2)
35 Receive 3
36 Store 3 (IndAddr 4)
37 Compute Incr 2 0 2
38 Compute Incr 4 0 4
39 ReadInstr (IndAddr 2)
40 Receive 3
41 Store 3 (IndAddr 4)
42 Compute Incr 2 0 2
43 Compute Incr 4 0 4
44 Compute Decr 5 0 5
45 Jump (Rel (-18))
```

```
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Load (ImmValue 1) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
ComputeI Gt 5 0 6
Branch 6 (Rel 7)
Load (IndAddr 2) 3
Compute Add 7 5 6
Store 3 (IndAddr 6)
Compute Incr 5 0 5
ComputeI Add 2 3 2
Jump (Rel (-7))
Compute Add 7 0 4
ComputeI Add 4 1 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 43) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 44) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 41) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 42) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 39) 2
TestAndSet (IndAddr 2)
```

```
94  Receive 3
95  Branch 3 (Rel 2)
96  Jump (Rel (-4))
97  Load (ImmValue 40) 4
98  ReadInstr (IndAddr 4)
99  Receive 5
100 Push 5
101 WriteInstr 0 (IndAddr 2)
102 Load (ImmValue 37) 2
103 TestAndSet (IndAddr 2)
104 Receive 3
105 Branch 3 (Rel 2)
106 Jump (Rel (-4))
107 Load (ImmValue 38) 4
108 ReadInstr (IndAddr 4)
109 Receive 5
110 Push 5
111 WriteInstr 0 (IndAddr 2)
112 Load (ImmValue 35) 2
113 TestAndSet (IndAddr 2)
114 Receive 3
115 Branch 3 (Rel 2)
116 Jump (Rel (-4))
117 Load (ImmValue 36) 4
118 ReadInstr (IndAddr 4)
119 Receive 5
120 Push 5
121 WriteInstr 0 (IndAddr 2)
122 Load (ImmValue 33) 2
123 TestAndSet (IndAddr 2)
124 Receive 3
125 Branch 3 (Rel 2)
126 Jump (Rel (-4))
127 Load (ImmValue 34) 4
128 ReadInstr (IndAddr 4)
129 Receive 5
130 Push 5
131 WriteInstr 0 (IndAddr 2)
132 Pop 6
133 PrintOut 6
134 Pop 6
135 PrintOut 6
136 Pop 6
137 PrintOut 6
138 Pop 6
139 PrintOut 6
140 Pop 6
141 PrintOut 6
```

```
142 Pop 6
143 PrintOut 6
144 Load (IndAddr 7) 7
145 Load (ImmValue 0) 2
146 Compute Sub 7 2 2
147 ComputeI Add 0 1 5
148 ComputeI Gt 5 0 6
149 Branch 6 (Rel 23)
150 Compute Add 7 5 6
151 Load (IndAddr 6) 4
152 Load (IndAddr 2) 3
153 Compute Lt 3 0 6
154 Branch 6 (Rel 2)
155 Store 4 (IndAddr 3)
156 Compute Incr 2 0 2
157 Load (IndAddr 2) 3
158 Compute Lt 3 0 6
159 Branch 6 (Rel 10)
160 Compute Add 3 0 6
161 TestAndSet (IndAddr 6)
162 Receive 6
163 Branch 6 (Rel 2)
164 Jump (Rel (-4))
165 ComputeI Add 3 1 3
166 WriteInstr 4 (IndAddr 3)
167 ComputeI Sub 3 1 3
168 WriteInstr 0 (IndAddr 3)
169 Compute Incr 5 0 5
170 ComputeI Add 2 2 2
171 Jump (Rel (-23))
172 Compute Decr 7 0 2
173 Load (IndAddr 2) 6
174 Load (IndAddr 7) 7
175 Jump (Ind 6)
176 Load (ImmValue 1) 2
177 Compute Sub 7 2 2
178 Load (ImmValue 1) 5
179 ComputeI Gt 5 0 6
180 Branch 6 (Rel 7)
181 Load (IndAddr 2) 3
182 Compute Add 7 5 6
183 Store 3 (IndAddr 6)
184 Compute Incr 5 0 5
185 ComputeI Add 2 3 2
186 Jump (Rel (-7))
187 Compute Add 7 0 4
188 ComputeI Add 4 1 4
189 Store 7 (IndAddr 4)
```

```
Compute Add 4 0 7
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 39) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 40) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 41) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 42) 4
ReadInstr (IndAddr 4)
```

```
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Load (ImmValue 43) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 44) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Pop 6
PrintOut 6
Load (IndAddr 7) 7
Load (ImmValue 0) 2
Compute Sub 7 2 2
ComputeI Add 0 1 5
ComputeI Gt 5 0 6
Branch 6 (Rel 23)
Compute Add 7 5 6
Load (IndAddr 6) 4
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 2)
Store 4 (IndAddr 3)
Compute Incr 2 0 2
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 10)
Compute Add 3 0 6
TestAndSet (IndAddr 6)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-4))
ComputeI Add 3 1 3
WriteInstr 4 (IndAddr 3)
```

```
286 ComputeI Sub 3 1 3
287 WriteInstr 0 (IndAddr 3)
288 Compute Incr 5 0 5
289 ComputeI Add 2 2 2
290 Jump (Rel (-23))
291 Compute Decr 7 0 2
292 Load (IndAddr 2) 6
293 Load (IndAddr 7) 7
294 Jump (Ind 6)
295 Nop
296 Nop
297 Load (ImmValue 8) 6
298 Push 6
299 Pop 6
300 Load (ImmValue 33) 2
301 TestAndSet (IndAddr 2)
302 Receive 3
303 Branch 3 (Rel 2)
304 Jump (Rel (-3))
305 Load (ImmValue 34) 4
306 WriteInstr 6 (IndAddr 4)
307 WriteInstr 0 (IndAddr 2)
308 Load (ImmValue 9) 6
309 Push 6
310 Pop 6
311 Load (ImmValue 35) 2
312 TestAndSet (IndAddr 2)
313 Receive 3
314 Branch 3 (Rel 2)
315 Jump (Rel (-3))
316 Load (ImmValue 36) 4
317 WriteInstr 6 (IndAddr 4)
318 WriteInstr 0 (IndAddr 2)
319 Load (ImmValue 10) 6
320 Push 6
321 Pop 6
322 Load (ImmValue 37) 2
323 TestAndSet (IndAddr 2)
324 Receive 3
325 Branch 3 (Rel 2)
326 Jump (Rel (-3))
327 Load (ImmValue 38) 4
328 WriteInstr 6 (IndAddr 4)
329 WriteInstr 0 (IndAddr 2)
330 Load (ImmValue 11) 6
331 Push 6
332 Pop 6
333 Load (ImmValue 39) 2
```

```
334 TestAndSet (IndAddr 2)
335 Receive 3
336 Branch 3 (Rel 2)
337 Jump (Rel (-3))
338 Load (ImmValue 40) 4
339 WriteInstr 6 (IndAddr 4)
340 WriteInstr 0 (IndAddr 2)
341 Load (ImmValue 12) 6
342 Push 6
343 Pop 6
344 Load (ImmValue 41) 2
345 TestAndSet (IndAddr 2)
346 Receive 3
347 Branch 3 (Rel 2)
348 Jump (Rel (-3))
349 Load (ImmValue 42) 4
350 WriteInstr 6 (IndAddr 4)
351 WriteInstr 0 (IndAddr 2)
352 Load (ImmValue 13) 6
353 Push 6
354 Pop 6
355 Load (ImmValue 43) 2
356 TestAndSet (IndAddr 2)
357 Receive 3
358 Branch 3 (Rel 2)
359 Jump (Rel (-3))
360 Load (ImmValue 44) 4
361 WriteInstr 6 (IndAddr 4)
362 WriteInstr 0 (IndAddr 2)
363 TestAndSet (DirAddr 1)
364 Receive 6
365 Branch 6 (Rel 2)
366 Jump (Rel (-3))
367 Load (ImmValue 5) 4
368 Load (ImmValue 0) 5
369 WriteInstr 5 (DirAddr 4)
370 Load (ImmValue 57) 6
371 Push 6
372 Pop 5
373 WriteInstr 5 (DirAddr 3)
374 WriteInstr 0 (DirAddr 2)
375 Load (ImmValue 1) 3
376 ReadInstr (IndAddr 3)
377 Receive 6
378 Branch 6 (Rel 2)
379 Jump (Rel (-3))
380 TestAndSet (DirAddr 1)
381 Receive 6
```

```
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 5) 4
Load (ImmValue 0) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 176) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Compute Equal 0 1 6
Branch 6 (Rel 4)
Load (ImmValue 2) 2
PrintOut 2
EndProg
Load (ImmValue 30) 3
Load (ImmValue 0) 2
ReadInstr (IndAddr 3)
Receive 4
Compute Add 2 4 2
ComputeI NEq 3 33 6
Compute Incr 3 0 3
Branch 6 (Rel (-5))
Compute Equal 2 0 6
Branch 6 (Rel 2)
Jump (Rel (-10))
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 8
>>> 9
>>> 10
>>> 11
>>> 12
>>> 13
>>> 13
>>> 12
>>> 11
>>> 10
>>> 9
```

```
>>> 8
```

Nested Procedures

Source

```
procedure p0() {
    print(90);
    p1();
    print(91);
    p2();
    print(92);
    p3();
    print(93);
    p4();
    print(94);
}

procedure p1() {
    print(10);
    p2();
    print(12);
    p3();
    print(13);
    p4();
    print(14);
}

procedure p2() {
    print(20);
    p3();
    print(23);
    p4();
    print(24);
}

procedure p3() {
    print(30);
    p4();
    print(34);
}

procedure p4() {
    print(40);
}

p0();
```

Generated SprIL

```
0  Branch 1 (Rel 6)
1  TestAndSet (DirAddr 2)
2  Receive 6
3  Branch 6 (Rel 2)
4  Jump (Rel (-3))
5  Jump (Rel 489)
6  ReadInstr (DirAddr 0)
7  Receive 3
8  Compute Equal 3 0 6
9  Branch 6 (Rel 2)
10 EndProg
11 TestAndSet (DirAddr 2)
12 Receive 6
13 Branch 6 (Rel 2)
14 Jump (Rel (-8))
15 ComputeI Add 1 30 3
16 TestAndSet (IndAddr 3)
17 Receive 6
18 Branch 6 (Rel 2)
19 Jump (Rel (-3))
20 ReadInstr (DirAddr 3)
21 Receive 3
22 Push 3
23 ComputeI Add 7 1 4
24 ReadInstr (DirAddr 4)
25 Receive 5
26 Load (ImmValue 5) 2
27 Compute Equal 5 0 6
28 Branch 6 (Rel 18)
29 ReadInstr (IndAddr 2)
30 Receive 3
31 Store 3 (IndAddr 4)
32 Compute Incr 2 0 2
33 Compute Incr 4 0 4
34 ReadInstr (IndAddr 2)
35 Receive 3
36 Store 3 (IndAddr 4)
37 Compute Incr 2 0 2
38 Compute Incr 4 0 4
39 ReadInstr (IndAddr 2)
40 Receive 3
41 Store 3 (IndAddr 4)
42 Compute Incr 2 0 2
43 Compute Incr 4 0 4
44 Compute Decr 5 0 5
45 Jump (Rel (-18))
```

```
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Load (ImmValue 1) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
ComputeI Gt 5 0 6
Branch 6 (Rel 7)
Load (IndAddr 2) 3
Compute Add 7 5 6
Store 3 (IndAddr 6)
Compute Incr 5 0 5
ComputeI Add 2 3 2
Jump (Rel (-7))
Compute Add 7 0 4
ComputeI Add 4 1 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 90) 6
Push 6
Pop 6
PrintOut 6
Compute Add 7 0 4
ComputeI Add 4 1 4
Load (ImmValue 0) 5
Load (ImmValue 90) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 180) 6
Push 6
Pop 2
Jump (Ind 2)
Load (ImmValue 91) 6
Push 6
Pop 6
PrintOut 6
```

```
Compute Add 7 0 4
ComputeI Add 4 1 4
Load (ImmValue 0) 5
Load (ImmValue 108) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 285) 6
Push 6
Pop 2
Jump (Ind 2)
Load (ImmValue 92) 6
Push 6
Pop 6
PrintOut 6
Compute Add 7 0 4
ComputeI Add 4 1 4
Load (ImmValue 0) 5
Load (ImmValue 126) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 372) 6
Push 6
Pop 2
Jump (Ind 2)
Load (ImmValue 93) 6
Push 6
Pop 6
PrintOut 6
Compute Add 7 0 4
ComputeI Add 4 1 4
Load (ImmValue 0) 5
Load (ImmValue 144) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 441) 6
Push 6
```

```
142 Pop 2
143 Jump (Ind 2)
144 Load (ImmValue 94) 6
145 Push 6
146 Pop 6
147 PrintOut 6
148 Load (IndAddr 7) 7
149 Load (ImmValue 0) 2
150 Compute Sub 7 2 2
151 ComputeI Add 0 1 5
152 ComputeI Gt 5 0 6
153 Branch 6 (Rel 23)
154 Compute Add 7 5 6
155 Load (IndAddr 6) 4
156 Load (IndAddr 2) 3
157 Compute Lt 3 0 6
158 Branch 6 (Rel 2)
159 Store 4 (IndAddr 3)
160 Compute Incr 2 0 2
161 Load (IndAddr 2) 3
162 Compute Lt 3 0 6
163 Branch 6 (Rel 10)
164 Compute Add 3 0 6
165 TestAndSet (IndAddr 6)
166 Receive 6
167 Branch 6 (Rel 2)
168 Jump (Rel (-4))
169 ComputeI Add 3 1 3
170 WriteInstr 4 (IndAddr 3)
171 ComputeI Sub 3 1 3
172 WriteInstr 0 (IndAddr 3)
173 Compute Incr 5 0 5
174 ComputeI Add 2 2 2
175 Jump (Rel (-23))
176 Compute Decr 7 0 2
177 Load (IndAddr 2) 6
178 Load (IndAddr 7) 7
179 Jump (Ind 6)
180 Load (ImmValue 1) 2
181 Compute Sub 7 2 2
182 Load (ImmValue 1) 5
183 ComputeI Gt 5 0 6
184 Branch 6 (Rel 7)
185 Load (IndAddr 2) 3
186 Compute Add 7 5 6
187 Store 3 (IndAddr 6)
188 Compute Incr 5 0 5
189 ComputeI Add 2 3 2
```

```
190 Jump (Rel (-7))
191 Compute Add 7 0 4
192 ComputeI Add 4 1 4
193 Store 7 (IndAddr 4)
194 Compute Add 4 0 7
195 Load (ImmValue 10) 6
196 Push 6
197 Pop 6
198 PrintOut 6
199 Compute Add 7 0 4
200 ComputeI Add 4 1 4
201 Load (ImmValue 0) 5
202 Load (ImmValue 213) 6
203 Push 6
204 Pop 5
205 Store 5 (IndAddr 4)
206 Compute Incr 4 0 4
207 Store 7 (IndAddr 4)
208 Compute Add 4 0 7
209 Load (ImmValue 285) 6
210 Push 6
211 Pop 2
212 Jump (Ind 2)
213 Load (ImmValue 12) 6
214 Push 6
215 Pop 6
216 PrintOut 6
217 Compute Add 7 0 4
218 ComputeI Add 4 1 4
219 Load (ImmValue 0) 5
220 Load (ImmValue 231) 6
221 Push 6
222 Pop 5
223 Store 5 (IndAddr 4)
224 Compute Incr 4 0 4
225 Store 7 (IndAddr 4)
226 Compute Add 4 0 7
227 Load (ImmValue 372) 6
228 Push 6
229 Pop 2
230 Jump (Ind 2)
231 Load (ImmValue 13) 6
232 Push 6
233 Pop 6
234 PrintOut 6
235 Compute Add 7 0 4
236 ComputeI Add 4 1 4
237 Load (ImmValue 0) 5
```

```
238 Load (ImmValue 249) 6
239 Push 6
240 Pop 5
241 Store 5 (IndAddr 4)
242 Compute Incr 4 0 4
243 Store 7 (IndAddr 4)
244 Compute Add 4 0 7
245 Load (ImmValue 441) 6
246 Push 6
247 Pop 2
248 Jump (Ind 2)
249 Load (ImmValue 14) 6
250 Push 6
251 Pop 6
252 PrintOut 6
253 Load (IndAddr 7) 7
254 Load (ImmValue 0) 2
255 Compute Sub 7 2 2
256 ComputeI Add 0 1 5
257 ComputeI Gt 5 0 6
258 Branch 6 (Rel 23)
259 Compute Add 7 5 6
260 Load (IndAddr 6) 4
261 Load (IndAddr 2) 3
262 Compute Lt 3 0 6
263 Branch 6 (Rel 2)
264 Store 4 (IndAddr 3)
265 Compute Incr 2 0 2
266 Load (IndAddr 2) 3
267 Compute Lt 3 0 6
268 Branch 6 (Rel 10)
269 Compute Add 3 0 6
270 TestAndSet (IndAddr 6)
271 Receive 6
272 Branch 6 (Rel 2)
273 Jump (Rel (-4))
274 ComputeI Add 3 1 3
275 WriteInstr 4 (IndAddr 3)
276 ComputeI Sub 3 1 3
277 WriteInstr 0 (IndAddr 3)
278 Compute Incr 5 0 5
279 ComputeI Add 2 2 2
280 Jump (Rel (-23))
281 Compute Decr 7 0 2
282 Load (IndAddr 2) 6
283 Load (IndAddr 7) 7
284 Jump (Ind 6)
285 Load (ImmValue 1) 2
```

```
286 Compute Sub 7 2 2
287 Load (ImmValue 1) 5
288 ComputeI Gt 5 0 6
289 Branch 6 (Rel 7)
290 Load (IndAddr 2) 3
291 Compute Add 7 5 6
292 Store 3 (IndAddr 6)
293 Compute Incr 5 0 5
294 ComputeI Add 2 3 2
295 Jump (Rel (-7))
296 Compute Add 7 0 4
297 ComputeI Add 4 1 4
298 Store 7 (IndAddr 4)
299 Compute Add 4 0 7
300 Load (ImmValue 20) 6
301 Push 6
302 Pop 6
303 PrintOut 6
304 Compute Add 7 0 4
305 ComputeI Add 4 1 4
306 Load (ImmValue 0) 5
307 Load (ImmValue 318) 6
308 Push 6
309 Pop 5
310 Store 5 (IndAddr 4)
311 Compute Incr 4 0 4
312 Store 7 (IndAddr 4)
313 Compute Add 4 0 7
314 Load (ImmValue 372) 6
315 Push 6
316 Pop 2
317 Jump (Ind 2)
318 Load (ImmValue 23) 6
319 Push 6
320 Pop 6
321 PrintOut 6
322 Compute Add 7 0 4
323 ComputeI Add 4 1 4
324 Load (ImmValue 0) 5
325 Load (ImmValue 336) 6
326 Push 6
327 Pop 5
328 Store 5 (IndAddr 4)
329 Compute Incr 4 0 4
330 Store 7 (IndAddr 4)
331 Compute Add 4 0 7
332 Load (ImmValue 441) 6
333 Push 6
```

```
Pop 2
Jump (Ind 2)
Load (ImmValue 24) 6
Push 6
Pop 6
PrintOut 6
Load (IndAddr 7) 7
Load (ImmValue 0) 2
Compute Sub 7 2 2
ComputeI Add 0 1 5
ComputeI Gt 5 0 6
Branch 6 (Rel 23)
Compute Add 7 5 6
Load (IndAddr 6) 4
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 2)
Store 4 (IndAddr 3)
Compute Incr 2 0 2
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 10)
Compute Add 3 0 6
TestAndSet (IndAddr 6)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-4))
ComputeI Add 3 1 3
WriteInstr 4 (IndAddr 3)
ComputeI Sub 3 1 3
WriteInstr 0 (IndAddr 3)
Compute Incr 5 0 5
ComputeI Add 2 2 2
Jump (Rel (-23))
Compute Decr 7 0 2
Load (IndAddr 2) 6
Load (IndAddr 7) 7
Jump (Ind 6)
Load (ImmValue 1) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
ComputeI Gt 5 0 6
Branch 6 (Rel 7)
Load (IndAddr 2) 3
Compute Add 7 5 6
Store 3 (IndAddr 6)
Compute Incr 5 0 5
ComputeI Add 2 3 2
```

```
382 Jump (Rel (-7))
383 Compute Add 7 0 4
384 ComputeI Add 4 1 4
385 Store 7 (IndAddr 4)
386 Compute Add 4 0 7
387 Load (ImmValue 30) 6
388 Push 6
389 Pop 6
390 PrintOut 6
391 Compute Add 7 0 4
392 ComputeI Add 4 1 4
393 Load (ImmValue 0) 5
394 Load (ImmValue 405) 6
395 Push 6
396 Pop 5
397 Store 5 (IndAddr 4)
398 Compute Incr 4 0 4
399 Store 7 (IndAddr 4)
400 Compute Add 4 0 7
401 Load (ImmValue 441) 6
402 Push 6
403 Pop 2
404 Jump (Ind 2)
405 Load (ImmValue 34) 6
406 Push 6
407 Pop 6
408 PrintOut 6
409 Load (IndAddr 7) 7
410 Load (ImmValue 0) 2
411 Compute Sub 7 2 2
412 ComputeI Add 0 1 5
413 ComputeI Gt 5 0 6
414 Branch 6 (Rel 23)
415 Compute Add 7 5 6
416 Load (IndAddr 6) 4
417 Load (IndAddr 2) 3
418 Compute Lt 3 0 6
419 Branch 6 (Rel 2)
420 Store 4 (IndAddr 3)
421 Compute Incr 2 0 2
422 Load (IndAddr 2) 3
423 Compute Lt 3 0 6
424 Branch 6 (Rel 10)
425 Compute Add 3 0 6
426 TestAndSet (IndAddr 6)
427 Receive 6
428 Branch 6 (Rel 2)
429 Jump (Rel (-4))
```

```
ComputeI Add 3 1 3
WriteInstr 4 (IndAddr 3)
ComputeI Sub 3 1 3
WriteInstr 0 (IndAddr 3)
Compute Incr 5 0 5
ComputeI Add 2 2 2
Jump (Rel (-23))
Compute Decr 7 0 2
Load (IndAddr 2) 6
Load (IndAddr 7) 7
Jump (Ind 6)
Load (ImmValue 1) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
ComputeI Gt 5 0 6
Branch 6 (Rel 7)
Load (IndAddr 2) 3
Compute Add 7 5 6
Store 3 (IndAddr 6)
Compute Incr 5 0 5
ComputeI Add 2 3 2
Jump (Rel (-7))
Compute Add 7 0 4
ComputeI Add 4 1 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 40) 6
Push 6
Pop 6
PrintOut 6
Load (IndAddr 7) 7
Load (ImmValue 0) 2
Compute Sub 7 2 2
ComputeI Add 0 1 5
ComputeI Gt 5 0 6
Branch 6 (Rel 23)
Compute Add 7 5 6
Load (IndAddr 6) 4
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 2)
Store 4 (IndAddr 3)
Compute Incr 2 0 2
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 10)
Compute Add 3 0 6
TestAndSet (IndAddr 6)
```

```
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-4))
ComputeI Add 3 1 3
WriteInstr 4 (IndAddr 3)
ComputeI Sub 3 1 3
WriteInstr 0 (IndAddr 3)
Compute Incr 5 0 5
ComputeI Add 2 2 2
Jump (Rel (-23))
Compute Decr 7 0 2
Load (IndAddr 2) 6
Load (IndAddr 7) 7
Jump (Ind 6)
Nop
Nop
Compute Add 7 0 4
ComputeI Add 4 1 4
Load (ImmValue 0) 5
Load (ImmValue 508) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 57) 6
Push 6
Pop 2
Jump (Ind 2)
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 90
>>> 10
>>> 20
>>> 30
>>> 40
>>> 34
>>> 23
>>> 40
>>> 24
>>> 12
>>> 30
>>> 40
```

```
>>> 34
>>> 13
>>> 40
>>> 14
>>> 91
>>> 20
>>> 30
>>> 40
>>> 34
>>> 23
>>> 40
>>> 24
>>> 92
>>> 30
>>> 40
>>> 34
>>> 93
>>> 40
>>> 94
```

Peterson

Source

```
global bool flag_0 = false;
global bool flag_1 = false;
global int turn = 0;
global int i = 0;

procedure p_0() {
    flag_0 = true;
    turn = 1;
    while ((flag_1 && (turn == 1))) {
        // wait
    }
    // begin critical section
    int j = 5;
    while ((j > 0)) {
        i = ++i;
        j = --j;
    }
    // end critical section
    flag_0 = false;
}

procedure p_1() {
    flag_1 = true;
    turn = 0;
```

```
25      while ((flag_0 && (turn == 0))) {
26          // wait
27      }
28      // begin critical section
29      int j = 5;
30      while ((j > 0)) {
31          i = --i;
32          j = --j;
33      }
34      // end critical section
35      flag_1 = false;
36  }
37
38  procedure test1(int j) {
39      while ((j > 0)) {
40          fork p_0();
41          fork p_1();
42          join;
43          print(i);
44
45          fork p_1();
46          fork p_0();
47          join;
48          print(i);
49
50          j = --j;
51      }
52  }
53
54  test1(10);
```

Generated SprIL

```
0   Compute Equal 1 0 6
1   Branch 6 (Rel 2)
2   Jump (Rel 7)
3   TestAndSet (DirAddr 2)
4   Receive 6
5   Branch 6 (Rel 2)
6   Jump (Rel (-3))
7   Load (ImmValue 0) 7
8   Jump (Rel 630)
9   ReadInstr (DirAddr 0)
10  Receive 3
11  Compute Equal 3 0 6
12  Branch 6 (Rel 2)
13  EndProg
14  TestAndSet (DirAddr 2)
```

```
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-8))
ComputeI Add 1 30 3
TestAndSet (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
ReadInstr (DirAddr 3)
Receive 3
Push 3
ComputeI Add 7 1 4
ReadInstr (DirAddr 4)
Receive 5
Load (ImmValue 5) 2
Compute Equal 5 0 6
Branch 6 (Rel 18)
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
Compute Decr 5 0 5
Jump (Rel (-18))
Load (ImmValue 57) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Load (ImmValue 1) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
```

```
63  ComputeI Gt 5 0 6
64  Branch 6 (Rel 7)
65  Load (IndAddr 2) 3
66  Compute Add 7 5 6
67  Store 3 (IndAddr 6)
68  Compute Incr 5 0 5
69  ComputeI Add 2 3 2
70  Jump (Rel (-7))
71  Compute Add 7 0 4
72  ComputeI Add 4 1 4
73  Store 7 (IndAddr 4)
74  Compute Add 4 0 7
75  Load (ImmValue 1) 6
76  Push 6
77  Load (ImmValue 33) 2
78  TestAndSet (IndAddr 2)
79  Receive 3
80  Branch 3 (Rel 2)
81  Jump (Rel (-4))
82  Load (ImmValue 34) 4
83  Pop 6
84  WriteInstr 6 (IndAddr 4)
85  WriteInstr 0 (IndAddr 2)
86  Pop 0
87  Load (ImmValue 1) 6
88  Push 6
89  Load (ImmValue 39) 2
90  TestAndSet (IndAddr 2)
91  Receive 3
92  Branch 3 (Rel 2)
93  Jump (Rel (-4))
94  Load (ImmValue 40) 4
95  Pop 6
96  WriteInstr 6 (IndAddr 4)
97  WriteInstr 0 (IndAddr 2)
98  Pop 0
99  Load (ImmValue 35) 2
100 TestAndSet (IndAddr 2)
101 Receive 3
102 Branch 3 (Rel 2)
103 Jump (Rel (-4))
104 Load (ImmValue 36) 4
105 ReadInstr (IndAddr 4)
106 Receive 5
107 Push 5
108 WriteInstr 0 (IndAddr 2)
109 Load (ImmValue 39) 2
110 TestAndSet (IndAddr 2)
```

```
111 Receive 3
112 Branch 3 (Rel 2)
113 Jump (Rel (-4))
114 Load (ImmValue 40) 4
115 ReadInstr (IndAddr 4)
116 Receive 5
117 Push 5
118 WriteInstr 0 (IndAddr 2)
119 Load (ImmValue 1) 6
120 Push 6
121 Pop 3
122 Pop 2
123 Compute Equal 2 3 4
124 Push 4
125 Pop 3
126 Pop 2
127 Compute And 2 3 4
128 Push 4
129 Pop 6
130 ComputeI Xor 6 1 6
131 Branch 6 (Rel 7)
132 Compute Add 7 0 4
133 ComputeI Add 4 1 4
134 Store 7 (IndAddr 4)
135 Compute Add 4 0 7
136 Load (IndAddr 7) 7
137 Jump (Rel (-38))
138 Load (ImmValue 5) 6
139 Push 6
140 Compute Add 7 0 6
141 ComputeI Add 6 1 6
142 Pop 5
143 Store 5 (IndAddr 6)
144 Compute Add 7 0 6
145 ComputeI Add 6 1 6
146 Load (IndAddr 6) 5
147 Push 5
148 Load (ImmValue 0) 6
149 Push 6
150 Pop 3
151 Pop 2
152 Compute Gt 2 3 4
153 Push 4
154 Pop 6
155 ComputeI Xor 6 1 6
156 Branch 6 (Rel 45)
157 Compute Add 7 0 4
158 ComputeI Add 4 2 4
```

```
159 Store 7 (IndAddr 4)
160 Compute Add 4 0 7
161 Load (ImmValue 37) 2
162 TestAndSet (IndAddr 2)
163 Receive 3
164 Branch 3 (Rel 2)
165 Jump (Rel (-4))
166 Load (ImmValue 38) 4
167 ReadInstr (IndAddr 4)
168 Receive 5
169 Push 5
170 WriteInstr 0 (IndAddr 2)
171 Pop 2
172 Compute Incr 2 0 4
173 Push 4
174 Load (ImmValue 37) 2
175 TestAndSet (IndAddr 2)
176 Receive 3
177 Branch 3 (Rel 2)
178 Jump (Rel (-4))
179 Load (ImmValue 38) 4
180 Pop 6
181 WriteInstr 6 (IndAddr 4)
182 WriteInstr 0 (IndAddr 2)
183 Pop 0
184 Compute Add 7 0 6
185 Load (IndAddr 6) 6
186 ComputeI Add 6 1 6
187 Load (IndAddr 6) 5
188 Push 5
189 Pop 2
190 Compute Decr 2 0 4
191 Push 4
192 Compute Add 7 0 6
193 Load (IndAddr 6) 6
194 ComputeI Add 6 1 6
195 Pop 2
196 Store 2 (IndAddr 6)
197 Push 2
198 Pop 0
199 Load (IndAddr 7) 7
200 Jump (Rel (-56))
201 Load (ImmValue 0) 6
202 Push 6
203 Load (ImmValue 33) 2
204 TestAndSet (IndAddr 2)
205 Receive 3
206 Branch 3 (Rel 2)
```

```
207 Jump (Rel (-4))
208 Load (ImmValue 34) 4
209 Pop 6
210 WriteInstr 6 (IndAddr 4)
211 WriteInstr 0 (IndAddr 2)
212 Pop 0
213 Load (IndAddr 7) 7
214 Load (ImmValue 0) 2
215 Compute Sub 7 2 2
216 ComputeI Add 0 1 5
217 ComputeI Gt 5 0 6
218 Branch 6 (Rel 23)
219 Compute Add 7 5 6
220 Load (IndAddr 6) 4
221 Load (IndAddr 2) 3
222 Compute Lt 3 0 6
223 Branch 6 (Rel 2)
224 Store 4 (IndAddr 3)
225 Compute Incr 2 0 2
226 Load (IndAddr 2) 3
227 Compute Lt 3 0 6
228 Branch 6 (Rel 10)
229 Compute Add 3 0 6
230 TestAndSet (IndAddr 6)
231 Receive 6
232 Branch 6 (Rel 2)
233 Jump (Rel (-4))
234 ComputeI Add 3 1 3
235 WriteInstr 4 (IndAddr 3)
236 ComputeI Sub 3 1 3
237 WriteInstr 0 (IndAddr 3)
238 Compute Incr 5 0 5
239 ComputeI Add 2 2 2
240 Jump (Rel (-23))
241 Compute Decr 7 0 2
242 Load (IndAddr 2) 6
243 Load (IndAddr 7) 7
244 Jump (Ind 6)
245 Load (ImmValue 1) 2
246 Compute Sub 7 2 2
247 Load (ImmValue 1) 5
248 ComputeI Gt 5 0 6
249 Branch 6 (Rel 7)
250 Load (IndAddr 2) 3
251 Compute Add 7 5 6
252 Store 3 (IndAddr 6)
253 Compute Incr 5 0 5
254 ComputeI Add 2 3 2
```

```
255 Jump (Rel (-7))
256 Compute Add 7 0 4
257 ComputeI Add 4 1 4
258 Store 7 (IndAddr 4)
259 Compute Add 4 0 7
260 Load (ImmValue 1) 6
261 Push 6
262 Load (ImmValue 35) 2
263 TestAndSet (IndAddr 2)
264 Receive 3
265 Branch 3 (Rel 2)
266 Jump (Rel (-4))
267 Load (ImmValue 36) 4
268 Pop 6
269 WriteInstr 6 (IndAddr 4)
270 WriteInstr 0 (IndAddr 2)
271 Pop 0
272 Load (ImmValue 0) 6
273 Push 6
274 Load (ImmValue 39) 2
275 TestAndSet (IndAddr 2)
276 Receive 3
277 Branch 3 (Rel 2)
278 Jump (Rel (-4))
279 Load (ImmValue 40) 4
280 Pop 6
281 WriteInstr 6 (IndAddr 4)
282 WriteInstr 0 (IndAddr 2)
283 Pop 0
284 Load (ImmValue 33) 2
285 TestAndSet (IndAddr 2)
286 Receive 3
287 Branch 3 (Rel 2)
288 Jump (Rel (-4))
289 Load (ImmValue 34) 4
290 ReadInstr (IndAddr 4)
291 Receive 5
292 Push 5
293 WriteInstr 0 (IndAddr 2)
294 Load (ImmValue 39) 2
295 TestAndSet (IndAddr 2)
296 Receive 3
297 Branch 3 (Rel 2)
298 Jump (Rel (-4))
299 Load (ImmValue 40) 4
300 ReadInstr (IndAddr 4)
301 Receive 5
302 Push 5
```

```
303 WriteInstr 0 (IndAddr 2)
304 Load (ImmValue 0) 6
305 Push 6
306 Pop 3
307 Pop 2
308 Compute Equal 2 3 4
309 Push 4
310 Pop 3
311 Pop 2
312 Compute And 2 3 4
313 Push 4
314 Pop 6
315 ComputeI Xor 6 1 6
316 Branch 6 (Rel 7)
317 Compute Add 7 0 4
318 ComputeI Add 4 1 4
319 Store 7 (IndAddr 4)
320 Compute Add 4 0 7
321 Load (IndAddr 7) 7
322 Jump (Rel (-38))
323 Load (ImmValue 5) 6
324 Push 6
325 Compute Add 7 0 6
326 ComputeI Add 6 1 6
327 Pop 5
328 Store 5 (IndAddr 6)
329 Compute Add 7 0 6
330 ComputeI Add 6 1 6
331 Load (IndAddr 6) 5
332 Push 5
333 Load (ImmValue 0) 6
334 Push 6
335 Pop 3
336 Pop 2
337 Compute Gt 2 3 4
338 Push 4
339 Pop 6
340 ComputeI Xor 6 1 6
341 Branch 6 (Rel 45)
342 Compute Add 7 0 4
343 ComputeI Add 4 2 4
344 Store 7 (IndAddr 4)
345 Compute Add 4 0 7
346 Load (ImmValue 37) 2
347 TestAndSet (IndAddr 2)
348 Receive 3
349 Branch 3 (Rel 2)
350 Jump (Rel (-4))
```

```
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Pop 2
Compute Decr 2 0 4
Push 4
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
Pop 6
WriteInstr 6 (IndAddr 4)
WriteInstr 0 (IndAddr 2)
Pop 0
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 2
Compute Decr 2 0 4
Push 4
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Pop 2
Store 2 (IndAddr 6)
Push 2
Pop 0
Load (IndAddr 7) 7
Jump (Rel (-56))
Load (ImmValue 0) 6
Push 6
Load (ImmValue 35) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 36) 4
Pop 6
WriteInstr 6 (IndAddr 4)
WriteInstr 0 (IndAddr 2)
Pop 0
Load (IndAddr 7) 7
```

```
Load (ImmValue 0) 2
Compute Sub 7 2 2
ComputeI Add 0 1 5
ComputeI Gt 5 0 6
Branch 6 (Rel 23)
Compute Add 7 5 6
Load (IndAddr 6) 4
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 2)
Store 4 (IndAddr 3)
Compute Incr 2 0 2
Load (IndAddr 2) 3
Compute Lt 3 0 6
Branch 6 (Rel 10)
Compute Add 3 0 6
TestAndSet (IndAddr 6)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-4))
ComputeI Add 3 1 3
WriteInstr 4 (IndAddr 3)
ComputeI Sub 3 1 3
WriteInstr 0 (IndAddr 3)
Compute Incr 5 0 5
ComputeI Add 2 2 2
Jump (Rel (-23))
Compute Decr 7 0 2
Load (IndAddr 2) 6
Load (IndAddr 7) 7
Jump (Ind 6)
Load (ImmValue 4) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
ComputeI Gt 5 1 6
Branch 6 (Rel 7)
Load (IndAddr 2) 3
Compute Add 7 5 6
Store 3 (IndAddr 6)
Compute Incr 5 0 5
ComputeI Add 2 3 2
Jump (Rel (-7))
Compute Add 7 0 4
ComputeI Add 4 2 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Compute Add 7 0 6
Load (IndAddr 6) 6
```

```
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Load (ImmValue 0) 6
Push 6
Pop 3
Pop 2
Compute Gt 2 3 4
Push 4
Pop 6
ComputeI Xor 6 1 6
Branch 6 (Rel 148)
Compute Add 7 0 4
ComputeI Add 4 1 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 5) 4
Load (ImmValue 0) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 60) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 5) 4
Load (ImmValue 0) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 245) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
```

```
Branch 6 (Rel 2)
Jump (Rel (-3))
Compute Equal 0 1 6
Branch 6 (Rel 4)
Load (ImmValue 2) 2
PrintOut 2
EndProg
Load (ImmValue 30) 3
Load (ImmValue 0) 2
ReadInstr (IndAddr 3)
Receive 4
Compute Add 2 4 2
ComputeI NEq 3 33 6
Compute Incr 3 0 3
Branch 6 (Rel (-5))
Compute Equal 2 0 6
Branch 6 (Rel 2)
Jump (Rel (-10))
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Pop 6
PrintOut 6
TestAndSet (DirAddr 1)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 5) 4
Load (ImmValue 0) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 245) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
TestAndSet (DirAddr 1)
```

```
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Load (ImmValue 5) 4
Load (ImmValue 0) 5
WriteInstr 5 (DirAddr 4)
Load (ImmValue 60) 6
Push 6
Pop 5
WriteInstr 5 (DirAddr 3)
WriteInstr 0 (DirAddr 2)
Load (ImmValue 1) 3
ReadInstr (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Compute Equal 0 1 6
Branch 6 (Rel 4)
Load (ImmValue 2) 2
PrintOut 2
EndProg
Load (ImmValue 30) 3
Load (ImmValue 0) 2
ReadInstr (IndAddr 3)
Receive 4
Compute Add 2 4 2
ComputeI NEq 3 33 6
Compute Incr 3 0 3
Branch 6 (Rel (-5))
Compute Equal 2 0 6
Branch 6 (Rel 2)
Jump (Rel (-10))
Load (ImmValue 37) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 38) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Pop 6
PrintOut 6
Compute Add 7 0 6
Load (IndAddr 6) 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
```

```
591 Load (IndAddr 6) 5
592 Push 5
593 Pop 2
594 Compute Decr 2 0 4
595 Push 4
596 Compute Add 7 0 6
597 Load (IndAddr 6) 6
598 Load (IndAddr 6) 6
599 ComputeI Add 6 1 6
600 Pop 2
601 Store 2 (IndAddr 6)
602 Push 2
603 Pop 0
604 Load (IndAddr 7) 7
605 Jump (Rel (-160))
606 Load (IndAddr 7) 7
607 Load (ImmValue 3) 2
608 Compute Sub 7 2 2
609 ComputeI Add 0 1 5
610 ComputeI Gt 5 1 6
611 Branch 6 (Rel 23)
612 Compute Add 7 5 6
613 Load (IndAddr 6) 4
614 Load (IndAddr 2) 3
615 Compute Lt 3 0 6
616 Branch 6 (Rel 2)
617 Store 4 (IndAddr 3)
618 Compute Incr 2 0 2
619 Load (IndAddr 2) 3
620 Compute Lt 3 0 6
621 Branch 6 (Rel 10)
622 Compute Add 3 0 6
623 TestAndSet (IndAddr 6)
624 Receive 6
625 Branch 6 (Rel 2)
626 Jump (Rel (-4))
627 ComputeI Add 3 1 3
628 WriteInstr 4 (IndAddr 3)
629 ComputeI Sub 3 1 3
630 WriteInstr 0 (IndAddr 3)
631 Compute Incr 5 0 5
632 ComputeI Add 2 2 2
633 Jump (Rel (-23))
634 Compute Decr 7 0 2
635 Load (IndAddr 2) 6
636 Load (IndAddr 7) 7
637 Jump (Ind 6)
638 Load (ImmValue 0) 6
```

```
639 Push 6
640 Pop 6
641 Load (ImmValue 33) 2
642 TestAndSet (IndAddr 2)
643 Receive 3
644 Branch 3 (Rel 2)
645 Jump (Rel (-3))
646 Load (ImmValue 34) 4
647 WriteInstr 6 (IndAddr 4)
648 WriteInstr 0 (IndAddr 2)
649 Load (ImmValue 0) 6
650 Push 6
651 Pop 6
652 Load (ImmValue 35) 2
653 TestAndSet (IndAddr 2)
654 Receive 3
655 Branch 3 (Rel 2)
656 Jump (Rel (-3))
657 Load (ImmValue 36) 4
658 WriteInstr 6 (IndAddr 4)
659 WriteInstr 0 (IndAddr 2)
660 Load (ImmValue 0) 6
661 Push 6
662 Pop 6
663 Load (ImmValue 39) 2
664 TestAndSet (IndAddr 2)
665 Receive 3
666 Branch 3 (Rel 2)
667 Jump (Rel (-3))
668 Load (ImmValue 40) 4
669 WriteInstr 6 (IndAddr 4)
670 WriteInstr 0 (IndAddr 2)
671 Load (ImmValue 0) 6
672 Push 6
673 Pop 6
674 Load (ImmValue 37) 2
675 TestAndSet (IndAddr 2)
676 Receive 3
677 Branch 3 (Rel 2)
678 Jump (Rel (-3))
679 Load (ImmValue 38) 4
680 WriteInstr 6 (IndAddr 4)
681 WriteInstr 0 (IndAddr 2)
682 Load (ImmValue 10) 6
683 Push 6
684 Compute Add 7 0 4
685 ComputeI Add 4 1 4
686 Load (ImmValue 1) 5
```

```
Pop 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue (-1)) 3
Store 3 (IndAddr 4)
Compute Incr 4 0 4
Load (ImmValue 707) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 430) 6
Push 6
Pop 2
Jump (Ind 2)
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
>>> 0
```

Recursion

Source

```
procedure rec(int i) {
    if ((i < 3)) {
        print(i);
        i = (i + 1);
        rec(i);
    } else {
        print(i);
    }
}

int i = 0;
rec(i);
```

Generated SprIL

```
Branch 1 (Rel 6)
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Jump (Rel 194)
ReadInstr (DirAddr 0)
Receive 3
Compute Equal 3 0 6
Branch 6 (Rel 2)
EndProg
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-8))
ComputeI Add 1 30 3
TestAndSet (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
ReadInstr (DirAddr 3)
Receive 3
Push 3
ComputeI Add 7 1 4
ReadInstr (DirAddr 4)
Receive 5
Load (ImmValue 5) 2
Compute Equal 5 0 6
Branch 6 (Rel 18)
```

```
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
Compute Decr 5 0 5
Jump (Rel (-18))
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Load (ImmValue 4) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
ComputeI Gt 5 1 6
Branch 6 (Rel 7)
Load (IndAddr 2) 3
Compute Add 7 5 6
Store 3 (IndAddr 6)
Compute Incr 5 0 5
ComputeI Add 2 3 2
Jump (Rel (-7))
Compute Add 7 0 4
ComputeI Add 4 2 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
```

```
77  Load (ImmValue 3) 6
78  Push 6
79  Pop 3
80  Pop 2
81  Compute Lt 2 3 4
82  Push 4
83  Pop 6
84  ComputeI Xor 6 1 6
85  Branch 6 (Rel 67)
86  Compute Add 7 0 4
87  ComputeI Add 4 1 4
88  Store 7 (IndAddr 4)
89  Compute Add 4 0 7
90  Compute Add 7 0 6
91  Load (IndAddr 6) 6
92  Load (IndAddr 6) 6
93  ComputeI Add 6 1 6
94  Load (IndAddr 6) 5
95  Push 5
96  Pop 6
97  PrintOut 6
98  Compute Add 7 0 6
99  Load (IndAddr 6) 6
100 Load (IndAddr 6) 6
101 ComputeI Add 6 1 6
102 Load (IndAddr 6) 5
103 Push 5
104 Load (ImmValue 1) 6
105 Push 6
106 Pop 3
107 Pop 2
108 Compute Add 2 3 4
109 Push 4
110 Compute Add 7 0 6
111 Load (IndAddr 6) 6
112 Load (IndAddr 6) 6
113 ComputeI Add 6 1 6
114 Pop 2
115 Store 2 (IndAddr 6)
116 Push 2
117 Pop 0
118 Compute Add 7 0 6
119 Load (IndAddr 6) 6
120 Load (IndAddr 6) 6
121 ComputeI Add 6 1 6
122 Load (IndAddr 6) 5
123 Push 5
124 Compute Add 7 0 4
```

```
125 ComputeI Add 4 1 4
126 Load (ImmValue 1) 5
127 Pop 3
128 Store 3 (IndAddr 4)
129 Compute Incr 4 0 4
130 Compute Add 7 0 6
131 Load (IndAddr 6) 6
132 Load (IndAddr 6) 6
133 ComputeI Add 6 1 6
134 Store 6 (IndAddr 4)
135 Compute Incr 4 0 4
136 Load (ImmValue (-1)) 3
137 Store 3 (IndAddr 4)
138 Compute Incr 4 0 4
139 Load (ImmValue 150) 6
140 Push 6
141 Pop 5
142 Store 5 (IndAddr 4)
143 Compute Incr 4 0 4
144 Store 7 (IndAddr 4)
145 Compute Add 4 0 7
146 Load (ImmValue 57) 6
147 Push 6
148 Pop 2
149 Jump (Ind 2)
150 Load (IndAddr 7) 7
151 Jump (Rel 14)
152 Compute Add 7 0 4
153 ComputeI Add 4 1 4
154 Store 7 (IndAddr 4)
155 Compute Add 4 0 7
156 Compute Add 7 0 6
157 Load (IndAddr 6) 6
158 Load (IndAddr 6) 6
159 ComputeI Add 6 1 6
160 Load (IndAddr 6) 5
161 Push 5
162 Pop 6
163 PrintOut 6
164 Load (IndAddr 7) 7
165 Load (IndAddr 7) 7
166 Load (ImmValue 3) 2
167 Compute Sub 7 2 2
168 ComputeI Add 0 1 5
169 ComputeI Gt 5 1 6
170 Branch 6 (Rel 23)
171 Compute Add 7 5 6
172 Load (IndAddr 6) 4
```

```
173 Load (IndAddr 2) 3
174 Compute Lt 3 0 6
175 Branch 6 (Rel 2)
176 Store 4 (IndAddr 3)
177 Compute Incr 2 0 2
178 Load (IndAddr 2) 3
179 Compute Lt 3 0 6
180 Branch 6 (Rel 10)
181 Compute Add 3 0 6
182 TestAndSet (IndAddr 6)
183 Receive 6
184 Branch 6 (Rel 2)
185 Jump (Rel (-4))
186 ComputeI Add 3 1 3
187 WriteInstr 4 (IndAddr 3)
188 ComputeI Sub 3 1 3
189 WriteInstr 0 (IndAddr 3)
190 Compute Incr 5 0 5
191 ComputeI Add 2 2 2
192 Jump (Rel (-23))
193 Compute Decr 7 0 2
194 Load (IndAddr 2) 6
195 Load (IndAddr 7) 7
196 Jump (Ind 6)
197 Nop
198 Nop
199 Load (ImmValue 0) 6
200 Push 6
201 Compute Add 7 0 6
202 ComputeI Add 6 1 6
203 Pop 5
204 Store 5 (IndAddr 6)
205 Compute Add 7 0 6
206 ComputeI Add 6 1 6
207 Load (IndAddr 6) 5
208 Push 5
209 Compute Add 7 0 4
210 ComputeI Add 4 2 4
211 Load (ImmValue 1) 5
212 Pop 3
213 Store 3 (IndAddr 4)
214 Compute Incr 4 0 4
215 Compute Add 7 0 6
216 ComputeI Add 6 1 6
217 Store 6 (IndAddr 4)
218 Compute Incr 4 0 4
219 Load (ImmValue (-1)) 3
220 Store 3 (IndAddr 4)
```

```
Compute Incr 4 0 4
Load (ImmValue 233) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 57) 6
Push 6
Pop 2
Jump (Ind 2)
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 0
>>> 1
>>> 2
>>> 3
```

Simple Concurrency

Source

```
global int num = 5;

procedure set_four() {
    (3+(2*(2*(2*332))));
    num = 4;
    print(num);
}

procedure set_six() {
    num = 6;
    print(num);
}

fork set_four();
fork set_six();
join;
print(num);
```

Generated SprIL

```
0  Branch 1 (Rel 6)
1  TestAndSet (DirAddr 2)
2  Receive 6
3  Branch 6 (Rel 2)
4  Jump (Rel (-3))
5  Jump (Rel 223)
6  ReadInstr (DirAddr 0)
7  Receive 3
8  Compute Equal 3 0 6
9  Branch 6 (Rel 2)
10 EndProg
11 TestAndSet (DirAddr 2)
12 Receive 6
13 Branch 6 (Rel 2)
14 Jump (Rel (-8))
15 ComputeI Add 1 30 3
16 TestAndSet (IndAddr 3)
17 Receive 6
18 Branch 6 (Rel 2)
19 Jump (Rel (-3))
20 ReadInstr (DirAddr 3)
21 Receive 3
22 Push 3
23 ComputeI Add 7 1 4
24 ReadInstr (DirAddr 4)
25 Receive 5
26 Load (ImmValue 5) 2
27 Compute Equal 5 0 6
28 Branch 6 (Rel 18)
29 ReadInstr (IndAddr 2)
30 Receive 3
31 Store 3 (IndAddr 4)
32 Compute Incr 2 0 2
33 Compute Incr 4 0 4
34 ReadInstr (IndAddr 2)
35 Receive 3
36 Store 3 (IndAddr 4)
37 Compute Incr 2 0 2
38 Compute Incr 4 0 4
39 ReadInstr (IndAddr 2)
40 Receive 3
41 Store 3 (IndAddr 4)
42 Compute Incr 2 0 2
43 Compute Incr 4 0 4
44 Compute Decr 5 0 5
45 Jump (Rel (-18))
```

```
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Load (ImmValue 1) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
ComputeI Gt 5 0 6
Branch 6 (Rel 7)
Load (IndAddr 2) 3
Compute Add 7 5 6
Store 3 (IndAddr 6)
Compute Incr 5 0 5
ComputeI Add 2 3 2
Jump (Rel (-7))
Compute Add 7 0 4
ComputeI Add 4 1 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 3) 6
Push 6
Load (ImmValue 2) 6
Push 6
Load (ImmValue 2) 6
Push 6
Load (ImmValue 2) 6
Push 6
Load (ImmValue 332) 6
Push 6
Pop 3
Pop 2
Compute Mul 2 3 4
Push 4
Pop 3
Pop 2
Compute Mul 2 3 4
Push 4
Pop 3
Pop 2
Compute Mul 2 3 4
Push 4
```

```
94  Pop 3
95  Pop 2
96  Compute Add 2 3 4
97  Push 4
98  Pop 0
99  Load (ImmValue 4) 6
100 Push 6
101 Load (ImmValue 33) 2
102 TestAndSet (IndAddr 2)
103 Receive 3
104 Branch 3 (Rel 2)
105 Jump (Rel (-4))
106 Load (ImmValue 34) 4
107 Pop 6
108 WriteInstr 6 (IndAddr 4)
109 WriteInstr 0 (IndAddr 2)
110 Pop 0
111 Load (ImmValue 33) 2
112 TestAndSet (IndAddr 2)
113 Receive 3
114 Branch 3 (Rel 2)
115 Jump (Rel (-4))
116 Load (ImmValue 34) 4
117 ReadInstr (IndAddr 4)
118 Receive 5
119 Push 5
120 WriteInstr 0 (IndAddr 2)
121 Pop 6
122 PrintOut 6
123 Load (IndAddr 7) 7
124 Load (ImmValue 0) 2
125 Compute Sub 7 2 2
126 ComputeI Add 0 1 5
127 ComputeI Gt 5 0 6
128 Branch 6 (Rel 23)
129 Compute Add 7 5 6
130 Load (IndAddr 6) 4
131 Load (IndAddr 2) 3
132 Compute Lt 3 0 6
133 Branch 6 (Rel 2)
134 Store 4 (IndAddr 3)
135 Compute Incr 2 0 2
136 Load (IndAddr 2) 3
137 Compute Lt 3 0 6
138 Branch 6 (Rel 10)
139 Compute Add 3 0 6
140 TestAndSet (IndAddr 6)
141 Receive 6
```

```
142 Branch 6 (Rel 2)
143 Jump (Rel (-4))
144 ComputeI Add 3 1 3
145 WriteInstr 4 (IndAddr 3)
146 ComputeI Sub 3 1 3
147 WriteInstr 0 (IndAddr 3)
148 Compute Incr 5 0 5
149 ComputeI Add 2 2 2
150 Jump (Rel (-23))
151 Compute Decr 7 0 2
152 Load (IndAddr 2) 6
153 Load (IndAddr 7) 7
154 Jump (Ind 6)
155 Load (ImmValue 1) 2
156 Compute Sub 7 2 2
157 Load (ImmValue 1) 5
158 ComputeI Gt 5 0 6
159 Branch 6 (Rel 7)
160 Load (IndAddr 2) 3
161 Compute Add 7 5 6
162 Store 3 (IndAddr 6)
163 Compute Incr 5 0 5
164 ComputeI Add 2 3 2
165 Jump (Rel (-7))
166 Compute Add 7 0 4
167 ComputeI Add 4 1 4
168 Store 7 (IndAddr 4)
169 Compute Add 4 0 7
170 Load (ImmValue 6) 6
171 Push 6
172 Load (ImmValue 33) 2
173 TestAndSet (IndAddr 2)
174 Receive 3
175 Branch 3 (Rel 2)
176 Jump (Rel (-4))
177 Load (ImmValue 34) 4
178 Pop 6
179 WriteInstr 6 (IndAddr 4)
180 WriteInstr 0 (IndAddr 2)
181 Pop 0
182 Load (ImmValue 33) 2
183 TestAndSet (IndAddr 2)
184 Receive 3
185 Branch 3 (Rel 2)
186 Jump (Rel (-4))
187 Load (ImmValue 34) 4
188 ReadInstr (IndAddr 4)
189 Receive 5
```

```
190 Push 5
191 WriteInstr 0 (IndAddr 2)
192 Pop 6
193 PrintOut 6
194 Load (IndAddr 7) 7
195 Load (ImmValue 0) 2
196 Compute Sub 7 2 2
197 ComputeI Add 0 1 5
198 ComputeI Gt 5 0 6
199 Branch 6 (Rel 23)
200 Compute Add 7 5 6
201 Load (IndAddr 6) 4
202 Load (IndAddr 2) 3
203 Compute Lt 3 0 6
204 Branch 6 (Rel 2)
205 Store 4 (IndAddr 3)
206 Compute Incr 2 0 2
207 Load (IndAddr 2) 3
208 Compute Lt 3 0 6
209 Branch 6 (Rel 10)
210 Compute Add 3 0 6
211 TestAndSet (IndAddr 6)
212 Receive 6
213 Branch 6 (Rel 2)
214 Jump (Rel (-4))
215 ComputeI Add 3 1 3
216 WriteInstr 4 (IndAddr 3)
217 ComputeI Sub 3 1 3
218 WriteInstr 0 (IndAddr 3)
219 Compute Incr 5 0 5
220 ComputeI Add 2 2 2
221 Jump (Rel (-23))
222 Compute Decr 7 0 2
223 Load (IndAddr 2) 6
224 Load (IndAddr 7) 7
225 Jump (Ind 6)
226 Nop
227 Nop
228 Load (ImmValue 5) 6
229 Push 6
230 Pop 6
231 Load (ImmValue 33) 2
232 TestAndSet (IndAddr 2)
233 Receive 3
234 Branch 3 (Rel 2)
235 Jump (Rel (-3))
236 Load (ImmValue 34) 4
237 WriteInstr 6 (IndAddr 4)
```

```
238 WriteInstr 0 (IndAddr 2)
239 TestAndSet (DirAddr 1)
240 Receive 6
241 Branch 6 (Rel 2)
242 Jump (Rel (-3))
243 Load (ImmValue 5) 4
244 Load (ImmValue 0) 5
245 WriteInstr 5 (DirAddr 4)
246 Load (ImmValue 57) 6
247 Push 6
248 Pop 5
249 WriteInstr 5 (DirAddr 3)
250 WriteInstr 0 (DirAddr 2)
251 Load (ImmValue 1) 3
252 ReadInstr (IndAddr 3)
253 Receive 6
254 Branch 6 (Rel 2)
255 Jump (Rel (-3))
256 TestAndSet (DirAddr 1)
257 Receive 6
258 Branch 6 (Rel 2)
259 Jump (Rel (-3))
260 Load (ImmValue 5) 4
261 Load (ImmValue 0) 5
262 WriteInstr 5 (DirAddr 4)
263 Load (ImmValue 155) 6
264 Push 6
265 Pop 5
266 WriteInstr 5 (DirAddr 3)
267 WriteInstr 0 (DirAddr 2)
268 Load (ImmValue 1) 3
269 ReadInstr (IndAddr 3)
270 Receive 6
271 Branch 6 (Rel 2)
272 Jump (Rel (-3))
273 Compute Equal 0 1 6
274 Branch 6 (Rel 4)
275 Load (ImmValue 2) 2
276 PrintOut 2
277 EndProg
278 Load (ImmValue 30) 3
279 Load (ImmValue 0) 2
280 ReadInstr (IndAddr 3)
281 Receive 4
282 Compute Add 2 4 2
283 ComputeI NEq 3 33 6
284 Compute Incr 3 0 3
285 Branch 6 (Rel (-5))
```

```
Compute Equal 2 0 6
Branch 6 (Rel 2)
Jump (Rel (-10))
Load (ImmValue 33) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 34) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Pop 6
PrintOut 6
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 4
>>> 6
>>> 6
```

Simple Procedures

Source

```
global int a = 3;

procedure p0(int c) {
    a = c;
    c = (c + 2);
}

int b = 2;
print(b); // should print 2

a = 1;

p0(b);
print(b); // should print 4
print(0);
print(a); // should print 2
```

Generated SprIL

```
0  Branch 1 (Rel 6)
1  TestAndSet (DirAddr 2)
2  Receive 6
3  Branch 6 (Rel 2)
4  Jump (Rel (-3))
5  Jump (Rel 134)
6  ReadInstr (DirAddr 0)
7  Receive 3
8  Compute Equal 3 0 6
9  Branch 6 (Rel 2)
10 EndProg
11 TestAndSet (DirAddr 2)
12 Receive 6
13 Branch 6 (Rel 2)
14 Jump (Rel (-8))
15 ComputeI Add 1 30 3
16 TestAndSet (IndAddr 3)
17 Receive 6
18 Branch 6 (Rel 2)
19 Jump (Rel (-3))
20 ReadInstr (DirAddr 3)
21 Receive 3
22 Push 3
23 ComputeI Add 7 1 4
24 ReadInstr (DirAddr 4)
25 Receive 5
26 Load (ImmValue 5) 2
27 Compute Equal 5 0 6
28 Branch 6 (Rel 18)
29 ReadInstr (IndAddr 2)
30 Receive 3
31 Store 3 (IndAddr 4)
32 Compute Incr 2 0 2
33 Compute Incr 4 0 4
34 ReadInstr (IndAddr 2)
35 Receive 3
36 Store 3 (IndAddr 4)
37 Compute Incr 2 0 2
38 Compute Incr 4 0 4
39 ReadInstr (IndAddr 2)
40 Receive 3
41 Store 3 (IndAddr 4)
42 Compute Incr 2 0 2
43 Compute Incr 4 0 4
44 Compute Decr 5 0 5
45 Jump (Rel (-18))
```

```
Load (ImmValue 54) 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Pop 2
WriteInstr 0 (DirAddr 1)
Jump (Ind 2)
ComputeI Add 1 30 3
WriteInstr 0 (IndAddr 3)
Jump (Abs 9)
Load (ImmValue 4) 2
Compute Sub 7 2 2
Load (ImmValue 1) 5
ComputeI Gt 5 1 6
Branch 6 (Rel 7)
Load (IndAddr 2) 3
Compute Add 7 5 6
Store 3 (IndAddr 6)
Compute Incr 5 0 5
ComputeI Add 2 3 2
Jump (Rel (-7))
Compute Add 7 0 4
ComputeI Add 4 2 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Load (ImmValue 31) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 32) 4
Pop 6
WriteInstr 6 (IndAddr 4)
WriteInstr 0 (IndAddr 2)
Pop 0
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Load (ImmValue 2) 6
Push 6
```

```
94  Pop 3
95  Pop 2
96  Compute Add 2 3 4
97  Push 4
98  Compute Add 7 0 6
99  Load (IndAddr 6) 6
100 ComputeI Add 6 1 6
101 Pop 2
102 Store 2 (IndAddr 6)
103 Push 2
104 Pop 0
105 Load (IndAddr 7) 7
106 Load (ImmValue 3) 2
107 Compute Sub 7 2 2
108 ComputeI Add 0 1 5
109 ComputeI Gt 5 1 6
110 Branch 6 (Rel 23)
111 Compute Add 7 5 6
112 Load (IndAddr 6) 4
113 Load (IndAddr 2) 3
114 Compute Lt 3 0 6
115 Branch 6 (Rel 2)
116 Store 4 (IndAddr 3)
117 Compute Incr 2 0 2
118 Load (IndAddr 2) 3
119 Compute Lt 3 0 6
120 Branch 6 (Rel 10)
121 Compute Add 3 0 6
122 TestAndSet (IndAddr 6)
123 Receive 6
124 Branch 6 (Rel 2)
125 Jump (Rel (-4))
126 ComputeI Add 3 1 3
127 WriteInstr 4 (IndAddr 3)
128 ComputeI Sub 3 1 3
129 WriteInstr 0 (IndAddr 3)
130 Compute Incr 5 0 5
131 ComputeI Add 2 2 2
132 Jump (Rel (-23))
133 Compute Decr 7 0 2
134 Load (IndAddr 2) 6
135 Load (IndAddr 7) 7
136 Jump (Ind 6)
137 Nop
138 Nop
139 Load (ImmValue 3) 6
140 Push 6
141 Pop 6
```

```
142 Load (ImmValue 31) 2
143 TestAndSet (IndAddr 2)
144 Receive 3
145 Branch 3 (Rel 2)
146 Jump (Rel (-3))
147 Load (ImmValue 32) 4
148 WriteInstr 6 (IndAddr 4)
149 WriteInstr 0 (IndAddr 2)
150 Load (ImmValue 2) 6
151 Push 6
152 Compute Add 7 0 6
153 ComputeI Add 6 1 6
154 Pop 5
155 Store 5 (IndAddr 6)
156 Compute Add 7 0 6
157 ComputeI Add 6 1 6
158 Load (IndAddr 6) 5
159 Push 5
160 Pop 6
161 PrintOut 6
162 Load (ImmValue 1) 6
163 Push 6
164 Load (ImmValue 31) 2
165 TestAndSet (IndAddr 2)
166 Receive 3
167 Branch 3 (Rel 2)
168 Jump (Rel (-4))
169 Load (ImmValue 32) 4
170 Pop 6
171 WriteInstr 6 (IndAddr 4)
172 WriteInstr 0 (IndAddr 2)
173 Pop 0
174 Compute Add 7 0 6
175 ComputeI Add 6 1 6
176 Load (IndAddr 6) 5
177 Push 5
178 Compute Add 7 0 4
179 ComputeI Add 4 2 4
180 Load (ImmValue 1) 5
181 Pop 3
182 Store 3 (IndAddr 4)
183 Compute Incr 4 0 4
184 Compute Add 7 0 6
185 ComputeI Add 6 1 6
186 Store 6 (IndAddr 4)
187 Compute Incr 4 0 4
188 Load (ImmValue (-1)) 3
189 Store 3 (IndAddr 4)
```

```
Compute Incr 4 0 4
Load (ImmValue 202) 6
Push 6
Pop 5
Store 5 (IndAddr 4)
Compute Incr 4 0 4
Store 7 (IndAddr 4)
Compute Add 4 0 7
Load (ImmValue 57) 6
Push 6
Pop 2
Jump (Ind 2)
Compute Add 7 0 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
PrintOut 6
Load (ImmValue 0) 6
Push 6
Pop 6
PrintOut 6
Load (ImmValue 31) 2
TestAndSet (IndAddr 2)
Receive 3
Branch 3 (Rel 2)
Jump (Rel (-4))
Load (ImmValue 32) 4
ReadInstr (IndAddr 4)
Receive 5
Push 5
WriteInstr 0 (IndAddr 2)
Pop 6
PrintOut 6
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 2
>>> 4
>>> 0
>>> 2
```

While

Source

```
int i = 100;
while ((i >= 0)) {
    1;
    print(i);
    i = (i - 1);
}
```

Generated SprIL

```
Branch 1 (Rel 6)
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
Jump (Rel 54)
ReadInstr (DirAddr 0)
Receive 3
Compute Equal 3 0 6
Branch 6 (Rel 2)
EndProg
TestAndSet (DirAddr 2)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-8))
ComputeI Add 1 30 3
TestAndSet (IndAddr 3)
Receive 6
Branch 6 (Rel 2)
Jump (Rel (-3))
ReadInstr (DirAddr 3)
Receive 3
Push 3
ComputeI Add 7 1 4
ReadInstr (DirAddr 4)
Receive 5
Load (ImmValue 5) 2
Compute Equal 5 0 6
Branch 6 (Rel 18)
ReadInstr (IndAddr 2)
Receive 3
Store 3 (IndAddr 4)
Compute Incr 2 0 2
Compute Incr 4 0 4
ReadInstr (IndAddr 2)
```

```
35 Receive 3
36 Store 3 (IndAddr 4)
37 Compute Incr 2 0 2
38 Compute Incr 4 0 4
39 ReadInstr (IndAddr 2)
40 Receive 3
41 Store 3 (IndAddr 4)
42 Compute Incr 2 0 2
43 Compute Incr 4 0 4
44 Compute Decr 5 0 5
45 Jump (Rel (-18))
46 Load (ImmValue 54) 5
47 Store 5 (IndAddr 4)
48 Compute Incr 4 0 4
49 Store 7 (IndAddr 4)
50 Compute Add 4 0 7
51 Pop 2
52 WriteInstr 0 (DirAddr 1)
53 Jump (Ind 2)
54 ComputeI Add 1 30 3
55 WriteInstr 0 (IndAddr 3)
56 Jump (Abs 9)
57 Nop
58 Nop
59 Load (ImmValue 100) 6
60 Push 6
61 Compute Add 7 0 6
62 ComputeI Add 6 1 6
63 Pop 5
64 Store 5 (IndAddr 6)
65 Compute Add 7 0 6
66 ComputeI Add 6 1 6
67 Load (IndAddr 6) 5
68 Push 5
69 Load (ImmValue 0) 6
70 Push 6
71 Pop 3
72 Pop 2
73 Compute GtE 2 3 4
74 Push 4
75 Pop 6
76 ComputeI Xor 6 1 6
77 Branch 6 (Rel 35)
78 Compute Add 7 0 4
79 ComputeI Add 4 2 4
80 Store 7 (IndAddr 4)
81 Compute Add 4 0 7
82 Load (ImmValue 1) 6
```

```
Push 6
Pop 0
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Pop 6
PrintOut 6
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Load (IndAddr 6) 5
Push 5
Load (ImmValue 1) 6
Push 6
Pop 3
Pop 2
Compute Sub 2 3 4
Push 4
Compute Add 7 0 6
Load (IndAddr 6) 6
ComputeI Add 6 1 6
Pop 2
Store 2 (IndAddr 6)
Push 2
Pop 0
Load (IndAddr 7) 7
Jump (Rel (-46))
Load (ImmValue 1) 2
WriteInstr 2 (DirAddr 0)
EndProg
```

Results

```
>>> 100
>>> 99
>>> 98
>>> 97
>>> 96
>>> 95
>>> 94
>>> 93
>>> 92
>>> 91
>>> 90
>>> 89
>>> 88
```

```
>>> 87
>>> 86
>>> 85
>>> 84
>>> 83
>>> 82
>>> 81
>>> 80
>>> 79
>>> 78
>>> 77
>>> 76
>>> 75
>>> 74
>>> 73
>>> 72
>>> 71
>>> 70
>>> 69
>>> 68
>>> 67
>>> 66
>>> 65
>>> 64
>>> 63
>>> 62
>>> 61
>>> 60
>>> 59
>>> 58
>>> 57
>>> 56
>>> 55
>>> 54
>>> 53
>>> 52
>>> 51
>>> 50
>>> 49
>>> 48
>>> 47
>>> 46
>>> 45
>>> 44
>>> 43
>>> 42
>>> 41
>>> 40
```

```
>>> 39
>>> 38
>>> 37
>>> 36
>>> 35
>>> 34
>>> 33
>>> 32
>>> 31
>>> 30
>>> 29
>>> 28
>>> 27
>>> 26
>>> 25
>>> 24
>>> 23
>>> 22
>>> 21
>>> 20
>>> 19
>>> 18
>>> 17
>>> 16
>>> 15
>>> 14
>>> 13
>>> 12
>>> 11
>>> 10
>>> 9
>>> 8
>>> 7
>>> 6
>>> 5
>>> 4
>>> 3
>>> 2
>>> 1
>>> 0
```