TEST THREE   **NAME**:
Type-checking and Stack Frames   **ID:**

QUESTION 1 (Typechecking)   (5 points)

a. The Modulus operator (%) was added to SPL in Coursework 2. The type behaviour of the new operator is shown by the following SPL code:

```
void main {
  float x.y;
  int z;
     y :=  12;
     x :=  (y+2) % 5;    // correct
     x := (x*7) % 5.5;   // error: wrong operand type
     x := 22.5 % 5;      // error: wrong operator type
     y := 15 % 7;        // error: type mismatch float -- int
}
```

Write the typechecking specification of the Modulus operator (I'm already giving you the abstract syntax tree representation).

Typecheck(Modulus(Exp e1, Exp e2), FunDecl f, Stable stable) =

QUESTION 2 (Typechecking)   (8 points)

Consider the following SPL program:

```
1 void main() {                    boolean g(int x, float y) {
2  int x; int y;                       // body is not important
3  boolean b;                      }
4  float f;
5  f := 16.5;
6  x := 40;
7  y := f + 1.5;
8  while (100 > x) {
9    b := g(x,y);
10   if (b) then {
11      print(x+f);  x := x + 2;
12   } else {
13      print(x+10); x := x + 5;
14  } }
15  print(z);
16 }
```

The **main**() function has four type errors (including undeclared variables). Indicate the line number where these errors appear. Briefly justify your answer.

QUESTION 3 (TPL)   (12 points)

Consider the following SPL program extract

```
int fone(int x, float y) {          boolean ftwo(float p1, int p2, float p3) {
 int a; int b;                          // body is not important
 float c;                           }
 boolean d;
 // body is not important
 d := ftwo(c,y,15.5);
 // more body comes here
 return c;
}
```

   a. Assuming there are enough registers for temporaries (even across calls)., write the stack frame layout for function `fone`. **[10 points]**

   b. How does `fone` access its parameter y? Write TPL code that stores 10.5 into y (from the body of `fone`). **[2 points]**