W3-Resubmission

Zeerak Waseem - csp265

14/12/2013

1 Exercise 1

1.1 a

When f(a, b, c) returns, it returns $7 + 3 + 12 = 22$. Thus when call-by-value is used, $r = 22$ x and y remain unchanged in the calling function. The reason is that call-by-value creates a copy of the variables to be used in the called function, thus leaving the variables unchanged in the calling function.
Prints: $r = 22, x = 5, y = 2$. Obviously it doesn't print the variable names, just their values.

1.2 b

When call-by-reference is used, the variables in the calling function are modified, thus x to be first 7 then subsequently changing x again to be 14. y is changed to 3.
Prints: $r = 33, x = 14, y = 5$.

1.3 c

When call-by-value-result is used, the variables are only changed when the called function returns, leaving the values in the calling function unchanged (as such, x would only change once rather than twice, as with call-by-reference. Thus x and y would be modified to 12 and 3 respectively.
Prints: $r = 22, x = 12, y = 3$.

2 Exercise 2

2.1 a

It would print 5 twice as g() has no access to the local $x's$, but only to the global.

2.2 b

It would print 4 and 9, as g() does have access to the local variables.

2.3 c

By editing in the `TpInterpret.sml` in the `callFun` function. By removing the symbol table, the variables would be referenced by the variables in the calling function, thus giving access to the local values.

3 Exercise 3

3.1 a

The type checker starts by matching on the equality, it then checks the left side of the expression, where an `int` is passed to the `chr()` function, which outputs a `char`, by means of the `read()` function. This gives us the knowledge that the right hand side is expected to be a `char`.

3.2 b