

Problem Set 3: Functions and Type-checking [Solutions]

Due October 16, 2020

Part I: Functions

For the following problem, consider the following program:

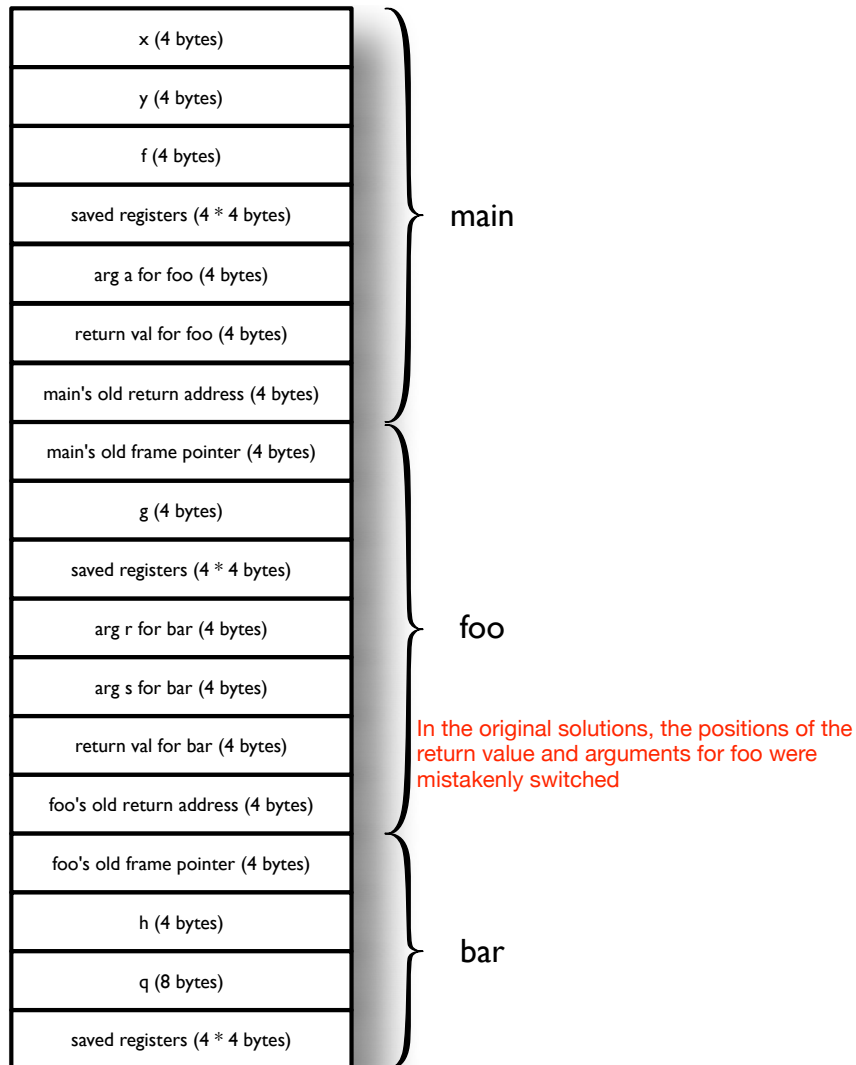
```
void main() {
    int x;
    float y;
    float f;
    ... //some computations
    f = foo(x + y);
    f += bar(x, y);
    ... //some computations
}

float foo(int a) {
    float g;
    g = bar(a, a);
    return g;
}

float bar(int r, int s) {
    float h;
    float q;
    q = 2.0;
    h = 1.0 * (r + s);
    if (r == s) {
        h = h * q;
    }
    return h;
}
```

1. Assume your program is running on a machine with 4 registers, using callee saves. Assume floats and ints are 4 bytes, and addresses are 4 bytes. Draw the *complete stack* (i.e., the stack including all active activation records) for the program *right after foo has called bar, and before bar returns*. For each slot in the stack, indicate what is stored there, and how much space that slot takes up.

Answer: [Assuming that bar actually returns float]

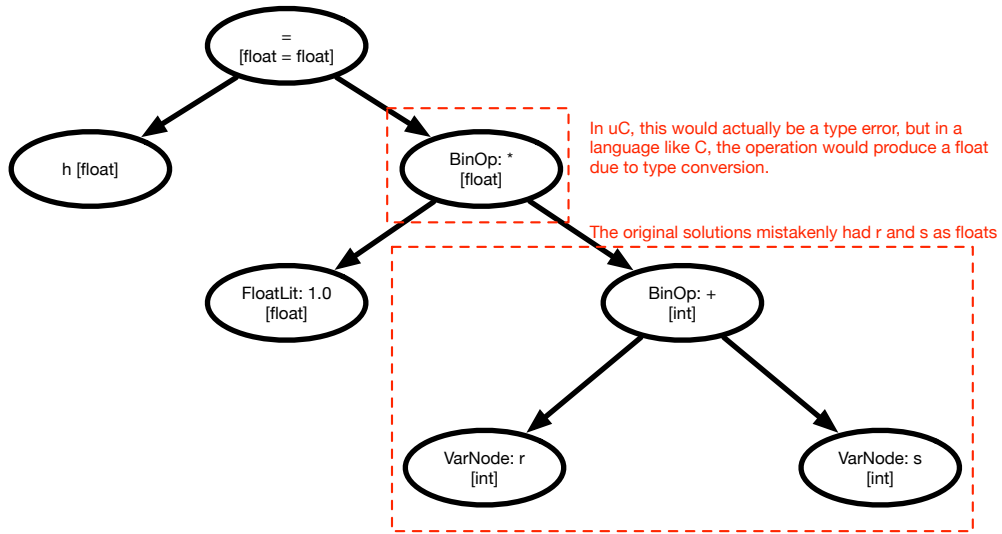


Part II: Type checking

For this part, consider the same code as in Part I.

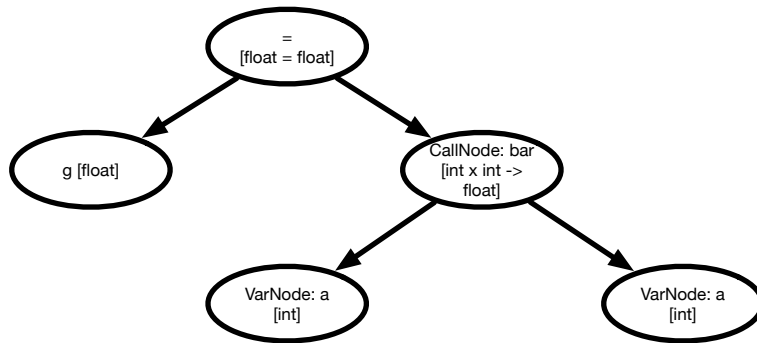
- Using the AST structure we developed for the course project, draw the AST for the line of code `h = 1.0 * (r + s)` and show on the AST you drew how type information is assigned to each node.

Answer:



2. Repeat that process for the line `g = bar(a, a)`

Answer:



3. Suppose the second parameter of `bar` was a `float` instead of an `int`. Mark with a star where type checking would fail in the previous problem.

Answer: [Problem node marked in red]

