University of Stirling

Computing Science and Mathematics

CSCU9A2 Tutorial 1 – Sample solutions Spring 2017

1. A high-level language is to help humans to write programs. Computers understand only machine language. Compilers are necessary to translate between these.

   The Java compiler translates into an intermediate language (bytecode). Thus it does not have to be specific to any particular hardware. A compiled Java program can be sent to (and run on) any computer, provided it has a Java Virtual Machine installed on it. This is particularly useful when communicating/distributing software via the Web.

   The JVM is a program which interprets a Java bytecode program (translates and executes, instruction-by-instruction).

2. We need variables as places to store data (e.g. numbers, text) in a program while it is running.

   Declarations inform the compiler of the variables that we intend using (with names and the types of data), so it can set up memory allocations.

   The compiler can spot mis-typed variable names, and can check consistent data usage such as not assigning a double value to an int variable, or using a String in arithmetic.

3. Step-by-step evaluation:

```
9 / 4   ->  2             ("primary school division")
9 % 4   ->  1             ("primary school remainder")
9 / 4.0   ->  2.25        (both operands converted to double)
2 * 5 / 2  -> 10 / 2  ->  5          (equal precedence, left to right)
5 / 2 * 2  -> 2 * 2  ->  4          (equal precedence, left to right)
2 * (5 / 2)  ->  2 * 2  ->  4       (brackets first, int division)
2 * (5 / 2.0)  ->  2 * 2.5  ->  5.0   (brackets first, double working)
3 * 5 - 2 * 2 - 4           3 * (5 - 2) * 2 - 4
  15 - 4 - 4                   3 * 3 * 2 - 4
     11 - 4                       9 * 2 - 4
        7                           18 - 4
                                       14
```

   Remember: `*`, `/` and `%` are high precedence and are carried out first; + and − are low precedence and are carried out second; work left to right if have equal precedence operators; parentheses override to give highest precedence. If in doubt, use parentheses to group as required.

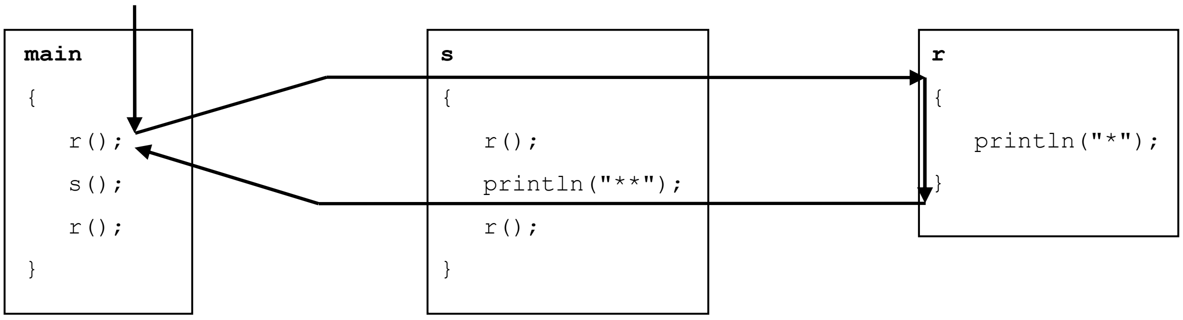
4. This is practice in keeping track of method calls/returns. Call r() is easy, and call s() is a bit harder!

   The output will be:

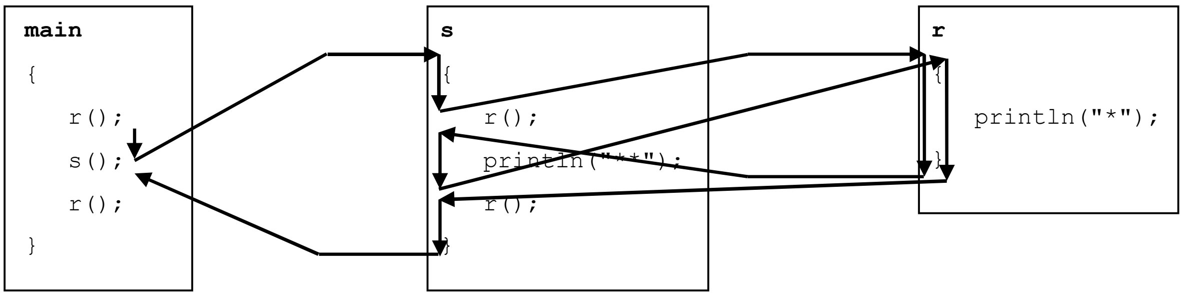
```
*                     produced by r();
*                     )
**                    )   produced by s();
*                     )
*                     produced by r();
```

   How it works: Here's the path of execution:

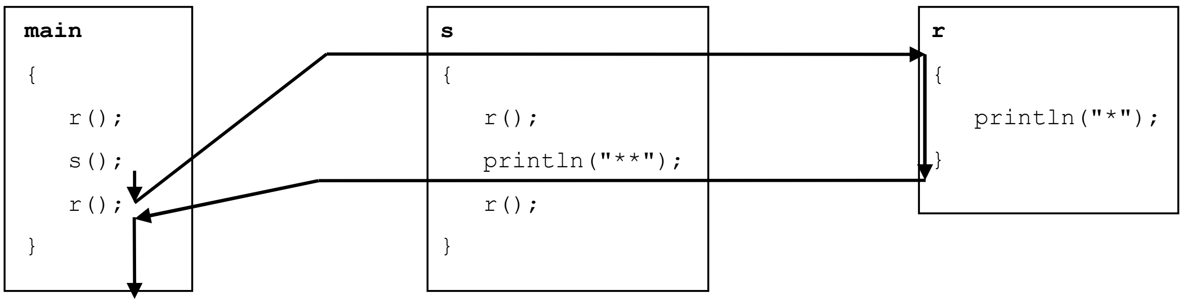
First there's a call to r:



Then there's a call to s:



Finally another call to r:



SBJ January 2017