

# Example Solutions for Programming Languages G22.2110 Summer 2007 hw08

Assigned Th 7/12/2007, due We 7/18/2007 at 1pm.

## How to Submit Homework Assignments

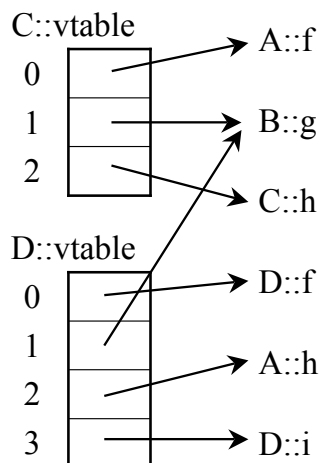
Email your answers, in either plain text format or as pdf, to Abhijit Guria <guria@cs.nyu.edu>. Assignments are due on Wednesdays at 1pm. This deadline will be strictly enforced.

## Reading Assignments

- For lecture on 7/12/2007: Scott 8.1, 8.2.0, 8.3.0-8.3.1, 8.5.0-8.5.2
- For lecture on 7/19/2007: Scott 7.2.4 (on CD);  
Cumming (<http://www.dcs.napier.ac.uk/course-notes/sml/manual.html>)

## Homework Assignments

1. Virtual method dispatch (10 points)  
Consider the following virtual method tables:



Write Java classes A, B, C, and D such that C and D have the vtables shown above. Make sure that your code causes no compiler errors.

2. SML (0 points)

Start teaching yourself SML by doing the following:

- 2a. If possible, find peers (other students who want to learn SML together with you) and gurus (people who already know SML, whom you can ask questions when you get stuck).
- 2b. Make sure you have access to the SML/NJ interpreter. Ask your guru if you are having problems with this step.
- 2c. Read the SML tutorial (see reading assignment above). Along the way, try things out with the SML interpreter that you installed in Step b.
- 2d. Familiarize yourself with the structure of the online SML documentation, so you can find information quickly when you need it. In particular, find the URLs for the Standard ML Basis Library and for the Harper tutorial.

3. Parameter passing modes (20 = 5 + 5 + 5 + 5 points)

Consider the following Ada program:

```
with Ada.Text_IO;
procedure Main is
  I : Integer := 1;
  A : array(1..2) of Integer := (1, 1);
  procedure P is begin
    Ada.Text_IO.Put_Line(
      Integer'Image(I) & Integer'Image(A(1)) & Integer'Image(A(2)));
  end P;
  procedure Q(J: in out Integer; K : in out Integer) is begin
    I := 2;
    K := 3;
    Ada.Text_IO.Put_Line(Integer'Image(J) & Integer'Image(K));
    P;
  end Q;
begin
  Q(I, A(I));
  P;
end Main;
```

- 3a. (5 points) What does the program print if “in out” means call by value?
- 3b. (5 points) What does the program print if “in out” means call by reference?
- 3c. (5 points) What does the program print if “in out” means call by value-result?
- 3d. (5 points) What does the program print if “in out” means call by name?

4. SML (10 = 2 + 3 + 2 + 3 points)

Continue teaching yourself SML by reading the example code from

<http://www.cs.nyu.edu/courses/summer07/G22.2110-001/hw08-sml-example.txt>

4a. (2 points) Run the program. Answer the prompts as follows:

```
Please enter a floating point number, or an empty line to end the list: 2.4
Please enter a floating point number, or an empty line to end the list: -2.1
Please enter a floating point number, or an empty line to end the list: 5.0
Please enter a floating point number, or an empty line to end the list: 4.23
Please enter a floating point number, or an empty line to end the list:
```

What does the program print?

4b. (3 points) What is the type of function `real_square`?

4c. (2 points) Remove the “: real” from the definition of function `real_square`.  
What error message does the compiler report?

4d. (3 points) Briefly explain the error message from Step c.

5. Parameter passing modes (10 = 3 + 7 points)

Consider the following C++ program:

```
#include <stdio.h>
void integer_divide(int x, int y, int& quotient, int& remainder) {
    quotient = x / y;
    remainder = x % y;
}
int main(int argc, char** argv) {
    int a=7, b=3, q, r;
    integer_divide(a, b, q, r);
    printf("%d / %d = %d + %d / %d\n", a, b, q, r, b);
    return 0;
}
```

5a. (3 points) What does this program print?

5b. (7 points) Translate the function `integer_divide` from C++ to C. The C version should have the signature

```
void integer_divide(int x, int y, int* quotient, int* remainder)
```

It should be line-by-line equivalent to the original C++ version.

---

<http://www.cs.nyu.edu/courses/summer07/G22.2110-001/hw08.pdf>

Total points: 50.