

Example Solutions for Programming Languages G22.2110 Summer 2007 hw07

Assigned Th 7/5/2007, due We 7/11/2007 at 1pm.

These are example solutions. Please keep in mind that often, there is not just one correct solution to a question. If you come up with different answers to the homework, then it may be that both your answers and these answers here are correct. Of course, these answers here may also contain mistakes; if you spot some, please let us know so we can correct them.

Reading Assignments

- For lecture on 7/5/2007: Scott 9.1, 9.4.3;
Sections from Java tutorial (<http://java.sun.com/docs/books/tutorial/java/TOC.html>)
about Classes, Objects, More on Classes, Interfaces, Inheritance, Strings, Packages
 - For lecture on 7/12/2007: Scott 8.1, 8.2.0, 8.3.0-8.3.1, 8.5.0-8.5.2
-

Homework Assignments

1. Java (12 = 2 + 3 + 3 + 4 points)

Download the code at the following URLs:

<http://www.cs.nyu.edu/courses/summer07/G22.2110-001/hw07-java-driver.txt>

<http://www.cs.nyu.edu/courses/summer07/G22.2110-001/hw07-java-intstack.txt>

<http://www.cs.nyu.edu/courses/summer07/G22.2110-001/hw07-java-intvector.txt>

Put them into files Driver.java, IntStack.java, and IntVector.java, respectively.

- 1a. (2 points) Run the driver. It should wait for your user input. Type in `push -55` and then `pop`. What is the output?

Example solutions

```
push -55
--> void
pop
--> -55
```

- 1b. (3 points) The “import” clauses of the Driver file import various system classes. What is the URL for class `InputStreamReader` in the Java 5.0 API specification?

Example solutions

<http://java.sun.com/j2se/1.5.0/docs/api/java/io/InputStreamReader.html>

- 1c. (3 points) Remove the “import InputStreamReader” clause from the driver, then change the code until it compiles again without that clause. What did you need to change?

Example solutions

```
before: BufferedReader in = new BufferedReader(  
        new      InputStreamReader(System.in));  
after:  BufferedReader in = new BufferedReader(  
        new java.io.InputStreamReader(System.in));
```

- 1d. (4 points) Which one of Ada’s “with” and “use” constructs corresponds to Java’s “import” construct? How does Java cope without the other construct?

Example solutions

- Ada’s “use” construct corresponds to Java’s “import” construct, since they both make entities from another package visible under their simple names, so no fully qualified names are required for using them.
- Ada’s “with” construct explicitly indicates that an Ada module depends on another module. In Java, these dependencies are implicit, in other words, the code does not need to mention them at the beginning. This means that the Java compiler will search files in the file system on demand for any identifiers that it can not resolve locally.

2. Java (6 = 3 + 3 points)

While you write the Java code for answering Question 3 below, you will probably get some error messages. Describe two error messages using the following format:

- Code: *a very short piece of code that triggers the error*
- Symptom: *the error message itself*
- Cause: *an explanation for what triggered the error message*
- Solution: *how to fix the code to prevent the error*

Example solutions

- first error
 - Code: `String userName = in.readLine();`
 - Symptom: `unreported exception java.io.IOException; must be caught or declared to be thrown`
 - Cause: At runtime, `readLine()` might fail due to I/O problems. The Java compiler expects that this behavior is specified at compile time, for example, by declaring that `main` can throw an `IOException`.
 - Solution: `public static void main(String[] args) throws IOException {`

```
- second error
- Code: float f = 3.141;
- Symptom: possible loss of precision, found: double, required: float
- Cause: literal floating point constants are treated as double numbers,
  i.e., 64-bit values. Writing them to a 32-bit float variable may lose
  precision. Therefore, Java requires an explicit cast.
- Solution: float f = (float)3.141;
```

3. Java (16 = 4 + 4 + 4 + 4 points)

Write Java 5 programs exercising the fundamental features.

3a. I/O (4 points)

Write a program that prompts the user for his or her name, reads the name from input, then politely greets the user by name. Here is an example interactive session:

```
What is your name?
Bob
Hello, Bob, nice to meet you!
```

Example solutions

```
import java.io.*;
class Main {
    public static void main(String[] args) throws IOException {
        System.out.println("What is your name?");
        BufferedReader in = new BufferedReader(new InputStreamReader(System.in));
        String line = in.readLine();
        System.out.println("Hello, " + line.trim() + ", nice to meet you!");
    }
}
```

3b. Libraries (4 points)

Write a program that uses Java 5 library functions to compute $\sqrt{2}$, $\sin(3.5)$, and $e^{2.5}$, and then prints the results like this (don't worry if the numbers are displayed in a slightly different format):

```
square root of 2.0:    1.4142135623730951
sine of 3.5:          -0.35078322768961984
e to the power of 2.5: 12.182493960703473
```

Example solutions

```
class Main {
    public static void main(String[] args) {
        double sqrt2 = Math.sqrt(2.0);
        System.out.println("square root of 2.0:    " + sqrt2);
        double sin35 = Math.sin(3.5);
        System.out.println("sine of 3.5:          " + sin35);
        double exp25 = Math.exp(2.5);
        System.out.println("e to the power of 2.5: " + exp25);
    }
}
```

```
}  
}
```

3c. Types (4 points)

The following code creates a variable `c` with the character value `'Z'`, and then prints a description and the value of the variable:

```
char c = 'Z';  
System.out.println("name c, type char, value " + c);
```

Extend this program by creating and printing more variables of different types. Your program should produce the following output:

```
name b, type boolean, value false  
name c, type char, value Z  
name f, type float, value 3.141  
name i, type int, value 42  
name ia, type int[], value { 1, 4, 9, 16 }  
name s, type String, value hello
```

Example solutions

```
class Main {  
    public static void main(String[] args) {  
        boolean b = false;  
        System.out.println("name b, type boolean, value " + b);  
        char c = 'Z';  
        System.out.println("name c, type char, value " + c);  
        float f = (float)3.141;  
        System.out.println("name f, type float, value " + f);  
        int i = 42;  
        System.out.println("name i, type int, value " + i);  
        int[] ia = { 1, 4, 9, 16 };  
        System.out.print("name ia, type int[], value {");  
        for (int j=0; j<ia.length; j++)  
            System.out.print((0 == j ? " " : ", ") + ia[j]);  
        System.out.println(" }");  
        String s = "hello";  
        System.out.println("name s, type String, value " + s);  
    }  
}
```

3d. Control flow (4 points)

Write a Java method `countOccurrences` that takes two parameters, a string and a character, and returns the number of occurrences of the character in the string. For example, `countOccurrences("hello", 'l')` should return 2.

Example solutions

```
static int countOccurrences(String s, char c) {  
    int result = 0;
```

```
    for (int i=0; i<s.length(); i++)
        if (s.charAt(i) == c)
            result++;
    return result;
}
```

4. Virtual method dispatch (16 = 4 + 4 + 4 + 4 points)

```
abstract class Animal {
    abstract String name();
    int legs() { return 4; }
    void move() { System.out.println("walking"); }
}
class Turtle extends Animal {
    String name() { return "turtle"; }
    void move() { System.out.println("swimming"); }
}
abstract class Bird extends Animal {
    int legs() { return 2; }
    void move() { System.out.println("flying"); }
}
class Swallow extends Bird {
    String name() { return "swallow"; }
}
class Ostrich extends Bird {
    String name() { return "ostrich"; }
    void move() { System.out.println("walking"); }
}
class Zoo {
    static void enumerate(Animal[] animals) {
        for (int i=0; i<animals.length; i++) {
            Animal a = animals[i];
            System.out.print("the " + a.name() + " has " + a.legs() + " legs; ");
            a.move();
        }
    }
    public static void main(String[] args) {
        Animal[] arr = new Animal[3];
        arr[0] = new Turtle();
        arr[1] = new Ostrich();
        arr[2] = new Swallow();
        enumerate(arr);
    }
}
```

4a. (4 points) What does this program print?

Example solutions

the turtle has 4 legs; swimming
the ostrich has 2 legs; walking
the swallow has 2 legs; flying

4b. (4 points) What is the vtable of class Turtle?

Example solutions

0: Turtle::name
1: Animal::legs
2: Turtle::move

4c. (4 points) What is the vtable of class Ostrich?

Example solutions

0: Ostrich::name
1: Bird::legs
2: Ostrich::move

4d. (4 points) What is the vtable of class Swallow?

Example solutions

0: Swallow::name
1: Bird::legs
2: Bird::move

<http://www.cs.nyu.edu/courses/summer07/G22.2110-001/hw07-example-solutions.pdf>

Total points: 50.