

600.107: Introduction to Programming

Exam 1

Peter H. Fröhlich
phf@cs.jhu.edu

June 7, 2005

Time: 40 Minutes

Start here: Please fill in the following important information using a **permanent pen** before you do **anything** else! Your exam will **not** be graded if you use a pencil or erasable ink on this page.

Name (print): _____

Email (print): _____

Cheat Sheet (yes/no): _____

Ethics Pledge: With your signature you **certify** the information above and you also **affirm** the following: “I agree to complete this exam without unauthorized assistance from any person, materials, or device.”

Signature: _____

Date: _____

Instructions: Please read these instructions carefully before you start. **Switch off** your phones, pagers, and other noisy gadgets! You are **not** allowed to have anything but a pen (pencil, eraser) and this exam on your desk. You are **not** allowed to talk to anyone during the exam. If you have a question, please raise your hand **quietly**. You must **remain seated quietly** until all exams have been collected. Remember that you can **not** claim grading errors if you do not use a **permanent** pen for your answers.

Do not open before you are told to do so!

You got _____ out of 40 points.

1 Binary Warmup

(10 points)

For each of the following statements, determine whether it is either **true** or **false**. (1 point each)

1. In contrast to software, hardware is easily adapted to new and unanticipated uses.
2. Unlike in math, variables in imperative programming are entities whose value can change over time.
3. In JAVA, the expression `a <= x <= b` yields a value of type `boolean`. (Assume all are `int`.)
4. In JAVA, the parentheses “(” and “)” have several different uses.
5. In JAVA, packages such as `java.util` and `java.lang` are the largest structuring unit.
6. Implicit type conversions are called “casts,” explicit type conversions are called “coercions.”
7. In JAVA, the expression `(int) 0.0 <= 12.0` leads to exactly one type conversion.
8. In JAVA, the loop `while (x > 0) { x = x-1; }` will always terminate (i.e. stop running).
9. In JAVA, the loop `while (x > 0) { x = x+1; }` will always terminate (i.e. stop running).
10. In JAVA, the loop `while (x < x+1) { x = x; }` will always terminate (i.e. stop running).

2 Tough Choices

(8 points)

For each of the following questions, circle **one** answer out of the choices given.

(2 points each)

- Which of the following is **true** about the programming language JAVA?
 - JAVA is a compiled language.
 - JAVA is an interpreted language.
 - JAVA is an object-oriented language.
 - All of the above.
 - None of the above.
- In JAVA, the expression $(1-4) \% 2$ evaluates to **what**?
 - 0
 - 1
 - 1
 - 3
 - None of the above.
- In JAVA, which of the following expressions evaluates to `true` for **all** possible integers `i`?
 - `i == 0`
 - `i < i + 1`
 - `i/2*2 == i`
 - All of the above.
 - None of the above.
- In JAVA, which of the following expressions is **equivalent** to $!(a \ || \ b)$? (Two expressions are equivalent iff they compute the identical result for all possible values of their constituent expressions.)
 - `!a && !b`
 - `(-a) || (-b)`
 - `!a || !b`
 - `a + b - true`
 - None of the above.

3 Short Answer

(8 points)

For each of the following questions, answer in **one to three** sentences, the shorter the better. (2 points each)

1. GRAHAM'S proposition: "*Nothing that doesn't end up in a line of your obituary is really worth doing.*" **Discuss!**
2. Describe in your own words **how** DIJKSTRA's Pebble Game for computing the GCD of two positive integers is played. Include a picture if that's helpful, but make sure that your description is **complete**. Do **not** use pieces of code, JAVA or otherwise!
3. Consider a JAVA program that contains the instruction `x = 2+y > (z/0);` which you are **sure** is executed. The program **compiles and runs** without errors. What you can say about the **types** that x, y, and z could possibly have? **Explain!**
4. Consider the **positive** integer number line including **zero**. You have **two** pebbles, one red, one blue. Describe your own Pebble Game for **adding** two numbers. You can **only** move each pebble up or down the number line **one step at a time**. Include a picture if that's helpful, but make sure that your description is **complete**. Do **not** use pieces of code, JAVA or otherwise!

4 Adding Things

(7 points)

Consider the following JAVA fragment. The program is supposed to read a sequence of **nonnegative** integers from the user and **add** them together, printing their sum. The sequence is terminated by a **negative** integer. **Fill in the blanks to make this work!**

```
public static void main( String[] args )
{
    Scanner console = new Scanner( in );
    int number = _____;
    _____ sum = 0;
    do
    {
        number = console._____( );
        if (number >= _____)
        {
            _____ = sum + number;
        }
    }
    while ( _____ >= _____ );
    out.println( sum );
}
```

5 Binary Madness

(7 points)

You can convert a **positive** integer i to its **binary** representation by successively dividing i by 2 until you get 0. The **remainder** of each division is 0 or 1, yielding the next binary digit. Write the **inside** of a JAVA `main` method that performs this computation: Read a positive integer from the user and print out its binary representation. You can **ignore** the fact that digits are produced in reverse order by this process!

This page is intentionally **mostly** blank in case you run out of space elsewhere. If you ended up here early, please go over **everything** again and remain seated **quietly**! Make sure that the title page is filled out correctly and in **permanent** pen. Maybe you want to "rewrite" your **answers** in permanent pen as well?