

UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE AND ENGINEERING
APS105 — Computer Fundamentals
Quiz #1
February 5, 1999, 4-5 p.m.

Exam Type A:

This is a "closed book" examination; no aids are permitted.

Calculator Type 2:

Non-programmable calculators only.

Write your answer in the space that follows each question. If more space is required, continue on the back of the page, and be sure to indicate on the front that you have done so.

The examination has 12 pages.

All questions are worth 5 marks each, except question 19 which is worth 10 marks.

For all questions, you may assume that the following StdIn methods are available:

StdIn.getInt(), StdIn.getDouble(), and StdIn.getChar().

Name _____

Circle your scheduled lab day: **Monday** or **Tuesday**

Student Number _____

MARKS

Question #	1	2	3	4	5	6	7	8	9	10
Marks										
Question #	11	12	13	14	15	16	17	18	19	
Marks										

Questions #1 and #2 refer to the Java program below:

```
class Hello
{
    public static void main (String[] args)
    {
        System.out.println ("Hello world!");
    }
}
```

Question 1 [5 marks]

If this program is compiled and executed, what will it print?

Question 2 [5 marks]

To be compiled and executed the program must be stored in a file. What should the name of the file be? What is the UNIX command sequence for compiling and executing the program?

Question 3 [5 marks]

Assume that in your home directory you have a directory called `assign1` and that `assign1` contains a file called `Test.java`. Give a sequence of UNIX commands to do the following:

1. set your working directory to your home directory;
2. create a directory called `assign2` in your home directory;
3. copy the file from `Test.java` in `assign1` to `assign2`;
4. list the contents of `assign2`.

Question 4 [5 marks]

What are the 3 essential components of a computer system?
Briefly describe the role of each component (about 5 words each).

For questions #5 to #10, indicate what the Java program will print if it is compiled and executed.

Question 5 [5 marks]

```
class A
{
    public static void main (String[] args)
    {
        System.out.print ("1");
        System.out.println ("t's true that");
        System.out.println (" 1");
        System.out.print (" +");
        System.out.print (" 1");
        System.out.println (" --");
        System.out.print (" 2");
        System.out.print (" Right?");
    }
}
```

Question 6 [5 marks]

```
class B
{
    public static void main (String[] args)
    {
        System.out.println ((6-15/3)+9*2/3/2);
        System.out.println (8/3);
        System.out.println (3.5/2);
    }
}
```

Question 7 [5 marks]

```
class C
{
    public static void main (String[] args)
    {
        double x;
        x = 8.5;
        System.out.println (Math.floor(x)/2);
    }
}
```

Question 8 [5 marks]

```
class D
{
    public static void main (String[] args)
    {
        int a,b;
        a = 23;
        b = 7;
        System.out.print ("The answer is ");
        if (a % b > 3);
            if (a != 3 && b == 7)
                System.out.println ("yes");
            else
                System.out.println ("no");
        }
    }
}
```

Question 9 [5 marks]

```
class E
{
    public static void main (String[] args)
    {
        int i;
        for (i=0; i < 5; i++);
            System.out.println("I must not forget semicolons.");
            System.out.println("I did not forget.");
        }
    }
}
```

Question 10 [5 marks]

```
class F
{
    public static void main (String[] args)
    {
        int i = 0;
        while (i < 10) {
            System.out.print (i);
            if (i == 4 || i == 9)
                System.out.println ();
            i++;
        }
    }
}
```

Question 11 [5 marks]

Rewrite the program in question #10 using a do loop instead of a while loop

```
class G
{
    public static void main (String[] args)
    {
        int i = 0;
    }
}
```

Question 12 [5 marks]

In the following Java program, there are between 5 and 10 errors. Find as many as you can and correct them in the *simplest* way you can think of.

```
class H {
    public static void main (String [] args) {
        int y;
        int a = StdIn.getInt(), b = StdIn.getInt();
        double c = Math.max(StdIn.getDouble,b);
        for( y=0; y<c; y+=Math.sqrt(a) )
        {
            if( a < y < c )
                if( b < a ! = a == 3 )
                    System.out.sqrt(c);
                else c -= a;
                else c += b;
                Math.floor( y );
            }
        }
    }
}
```

Question 13 [5 marks]

Consider the following Java variable declarations:

```
int i;
double x;
boolean b;
```

How many bits of storage does each variable require?

Question 14 [5 marks]

What does it mean if a Java *method* is declared as *static*? What are the restrictions on *static* methods? (30 words or less)

Question 15 [5 marks]

What is the difference between syntax and semantics of a programming language? (20 words or less)
What type of errors will a compiler, such as javac, always identify for you?

Question 16 [5 marks]

Complete the following Java program by writing a method `isLeapYear` that calculates whether a given year is a leap year (i.e. has 366 days) or not. All years that are evenly divisible by 4 are leap years, except the century years after 1582 that are evenly divisible by 100. However, of the century years after 1582, those evenly divisible by 400 are leap years. For example, 1900 is not a leap year but 1200, 1500, and 2000 are. The method should take an `integer` parameter, called `year`, and return a `boolean` result.

```
class W {
    public static void main (String [] args)
    {
        int y;
        while( true ) {
            System.out.print("Enter the year: ");
            y = StdIn.getInt();
            if( y < 0 ) break;
            System.out.print ("Is it a leap year? ");
            System.out.println( isLeapYear(y) );
        }
    }
}
```

Question 17 [5 marks]

Write a Java program that prompts for an integer n and prints out a right triangle pattern of "*" characters, with the two perpendicular sides of length n . An example of the execution of the program is shown below. Be sure to print your triangle in the *same orientation* as shown. The input typed by the user is shown in **bold**.

```
Enter an integer: 4
*
**
***
****

class X {
    public static void main(String[] args)
    {
```

```
    }
}
```

9

Question 18 [5 marks]

Write two Java methods, $Y.max(a, b)$ and $Y.min(a, b)$ which accept double parameters and return a double result corresponding to the maximum or minimum of a and b . *Do not* use any `if` statements, rather use `Math.abs()` and properties of the formula $|a - b| + a + b$ to compute your answer. *Do not write a main() method.*

```
class Y {
```

```
    }
```

10

Question 19 [10 marks]

Complete the following Java program *using objects*. The goal is to remember the number entered only if it has a larger fractional part than any other number entered so far. The testing program, and example output from the program, are shown below. The input typed by the user is shown in **bold**.

Write your class BigFraction on the next page. Inside the class, write any instance variables, a constructor, and the instance method, saveBiggerFraction (BigFraction), which remembers the *num-ber* with the larger fractional and prints it out.

```
public class BigFractionTest
{
    // this is the test program for class BigFraction
    public static void main (String[] args)
    {
        BigFraction big = new BigFraction ( 0.0 );
        while ( true ) {

            // get a double number from the user
            System.out.print("Enter a double number: ");
            double userInput = StdIn.getDouble();
            if ( userInput == 0.0 )
                break;

            // remember the number with the bigger fraction
            // don't forget to print it inside the method
            BigFraction other = new BigFraction( userInput );
            big.saveBiggerFraction( other );
        }
        System.out.println("Done.");
    }
}

Enter a double number: 5
0.0
Enter a double number: 3.25
3.25
Enter a double number: 1.3
1.3
Enter a double number: 5.1
1.3
Enter a double number: 3.3
1.3
Enter a double number: -1.7
-1.7
Enter a double number: 0
Done.
```

11

```
public class BigFraction {
```

```
}
```

12