

Midterm Solutions

PROBLEM 1 : (*Quick Ones (4 points)*)

- A. Give an example of a variable declaration.

```
int a;
```

- B. Given that you wrote and compiled an applet named Junk.java that uses DoubleFields, write the line of HTML code necessary to include your applet on a webpage.

```
<applet code="Junk.class" height=500 width=500 archive="awb.jar"></applet>
```

PROBLEM 2 : (*Parsing (12 points)*)

For the following lines of code, state whether each line is a legal syntactical statement. If it is not a legal Java statement, explain why. If it is a legal Java statement, show how to parse it using the grammar below.

1. <name> ==> any string of alphanumeric symbols that begins with a letter
2. <statement> ==> <name> = <expression> ;
3. <statement> ==> <name> = new <class>(<arguments>);
4. <statement> ==> <name>.<method>(<arguments>); | <method>(<arguments>);
5. <arguments> == possibly empty list of <expression>s separated by commas
6. <expression> ==> <string-expression> | <int-expression> | <oth-expression>
7. <string-expression> ==> <string-expression> + <string-expression>
8. <string-expression> ==> <string>
9. <string> ==> " any sequence of characters "
10. <string> ==> <name>
11. <int-expression> ==> <name>
12. <int-expression> ==> <int-expression> <op> <int-expression>
13. <int-expression> ==> <pos-int> | - <pos-int>
14. <pos-int> ==> \it any sequence of digits
15. <op> ==> + | - | * | / | %
16. <method> ==> setText | getText | getInt | setInt | add | actionPerformed
17. <class> ==> TextField | IntField | Button | ActionEvent

- A. `x = x + -1;`

```
#2 <name> = <expression>;
#6 <name> = <int-expression>;
#12 <name> = <int-expression> <op> <int-expression>;
#11 <name> = <name> <op> <int-expression>;
#13 <name> = <name> <op> -<pos-int>;
#1, #1 x = x <op> -<pos-int>;
#15 x = x + -<pos-int>;
#14 x = x + -1;
```

B. `s = b / "Hello";`

Not legal java syntax. / is is not a string operation

C. `a.setText("Peace" + "To all");`

```
#4 <name>.<method>( <arguments> );
#5 <name>.<method>( <expression> );
#6 <name>.<method>( <string-expression> );
#7 <name>.<method>( <string-expression> + <string-expression> );
#1 a.<method>( <string-expression> + <string-expression> );
#16 a.setText( <string-expression> + <string-expression> );
#8,#8 a.setText( <string> + <string> );
#9,#9 a.setText( "Peace" + "To all" );
```

PROBLEM 3 : (DoSomething a lot (10 points))

Janel Student receives the following specification for a problem

Write a subroutine called `DoIt` that takes integer one argument n and calls the subroutine `DoSomething` n times.

As an example, the code calling the subroutine looks like:

```
DoIt(4);
```

A. She then writes the following code, but she forgets to write a function declaration. Fill it in for her.

```
// FILL IN FUNCTION DECLARATION
\textbf{void DoIt(int n)}
{
    int i = 1;
    while (i < n)
    {
        DoSomething();
        i = i - 1;
    }
}
```

B. When Janel runs her code, it doesn't seem to "work." What will happen? What is wrong?

The program will never stop running. It is in an infinite loop.

C. What needs to be changed so that it now works according to the specification? In other words, change the code so that it now calls `DoSomething` exactly n times.

Change `i = i - 1` to `i = i + 1` and change `while (i < n)` to `while (i <= n)`

PROBLEM 4 : (What happens? (10 points))

The following is a portion of code of a Java program:

```
import awb.*;
import java.awt.*;
import java.awt.event.*;
```

```

public class CalcNum extends java.applet.Applet implements ActionListener
{
    double a,b,c;
    DoubleField aF, bF, cF;
    Button b;
    public void init()
    {
        aF= new DoubleField(10);
        aF.setLabel("Var A");
        bF= new DoubleField(10);
        bF.setLabel("Var B");
        cF= new DoubleField(10);
        cF.setLabel("Solution");
        b1 = new Button("Calculate");
        b1.addActionListener(this);
        add(aF);
        add(bF);
        add(b1);
        add(cF);

        aF.setDouble(3.0);
        bF.setDouble(5);
        c = 0;
    }

    public void actionPerformed(ActionEvent event)
    {
        Object cause = event.getSource();
        if (cause == b1)
        {
            a = aF.getDouble();
            b = bF.getDouble();
            c = c + a + a * b;
            cF.setDouble(c);
        }
    }
}

```

A. Draw what the applet looks like immediately after it is loaded.

Var A	3.0	Var B	5.0

Calculate	Solution		

B. The button labeled Calculate is pressed three times. Give the output for the field labeled Solution *after* each press.

18

36

54

PROBLEM 5: (Word Play (8 points))

After executing the following code, what are the values of the variables below:

```
int i = 3;
int a;
bool b;
String yes = "Stick to";
String no = "the rivers";
String s,t;
i = yes.length();
if (yes.indexOf("tic") > -1)
    s = yes.substring(0,4);
else
    s = "slimpy";

t = yes + no;
t = t.substring(6,t.length());
b = (t.length() > 0);
```

A. i 8

B. s "Stic"

C. t "Stick tothe rivers"

D. b true

PROBLEM 6: (Mean? (5 points))

Write a subroutine called average that takes two real numbers and returns the average of the two numbers. For example, if I call your subroutine as follows:

```
double a = 10.0;
double b = 20.0;
double c;
```

```
c = average(a,b);
```

then the value of c would be 15.0.

```
double average(double num1, double num2)
{
    return (num1+num2)/2.0;
}
```

PROBLEM 7 : (Cool Math Extra Credit (2 points))

What is

$$\frac{1}{1} + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \cdots + \frac{1}{2^n}$$

as $n \rightarrow \infty$?

2

What is

$$\frac{2 \cdot 2 \cdot 4 \cdot 4 \cdot 6 \cdot 6 \cdot 8 \cdots 2n}{3 \cdot 3 \cdot 5 \cdot 5 \cdot 7 \cdot 7 \cdots 2n - 1}$$

as $n \rightarrow \infty$? $\pi/4$

John Wallis (1616-1703) in around 1650 (he may have been predated or certainly at least contemporary with Japanese mathematicians who came up with the same idea) came up with the Wallis Formula which says that:

$$\frac{\pi}{2} = \frac{2 \cdot 2 \cdot 4 \cdot 4 \cdot 6 \cdot 6 \cdot 8 \cdots 2n}{3 \cdot 3 \cdot 5 \cdot 5 \cdot 7 \cdot 7 \cdots 2n - 1}$$

It was remarkable in that it was the first infinite series for pi that did not involve irrational numbers and was computed before calculus. It involves computing the area under a quadrant of a circle.