CS101 Midterm Exam, Spring 2009

February 26, 2009

Instructions: This exam is open book. You can use any reference books. But no computers are allowed. All the programs MUST be written using JAVA. If you do not know the answer, try to give your best guess. Partial credits will be given in the marking. The total marks are 100 points. There is also a bonus question worth of 25 points. You have 1 hour and 15 mins to complete the exam.

1 Program Tracing [25 points]

Trace the following program and write the output.

```java
int a = 1;
int b = 1;
int N = 2;
for(int n = 0; n <= N; n++) {
    int c = a + b;
    b = a;
    a = c;
}
System.out.println("a = " + a);
System.out.println("b = " + b);
```

2 Functions and Arguments [25 points]

What is the output of the following program? Justify your answer.

```java
public class Add {
    public static void add(int x, int y, int sum)
    {
        sum = x + y;
    }

    public static void main(String[] args)
    {
        int x = 1;
        int y = 2;
```
int sum = 0;
add(x, y, sum);
System.out.print(sum);
}
}

3 Sum of a Sequence [25 points]

Write a function to compute $1\times2 + 3\times4 + 5\times6 + \ldots + (2n-1)\times(2n)$ using while loop and for loop respectively. The function’s interface is as follows:

```java
public static int sumOfProd(int n)
```

—— CHOOSE EITHER QUESTION 4 OR QUESTION 5 ——

4 Standard Input and Output [25 points]

Write a class with a main method to read from a standard input, process the data and output the result. The input includes alternate `int` and `double` numbers, like

```
10 123.4
20 256.3
-30 0.02
...
```

Write a procedure to compute the sum of all the `negative` integers and the largest `double` number. Note that we do not know how many numbers are in the stream. We use `StdIn.readInt()` and `StdIn.readDouble()` to read the numbers from the input stream. To check whether there is still data coming in, use `StdIn.isEmpty()`. `StdIn.isEmpty()` returns `true` when there is no data in the stream. You can use the following loop in your program.

```java
while(!StdIn.isEmpty()) {
    int a = StdIn.readInt();
    double b = StdIn.readDouble();
    // process your data here
}
```

// output the results

5 Array [25 points]

Write a function to check whether array `a` includes all the numbers in array `b`. For example, `{1, 2, 3}` contains all the numbers in the array `{3, 2}`, but does not contain all the numbers in the array `{3, 4}`. The interface of the function is

```java
public static boolean hasAllNumbers(int[] a, int[] b)
```

[Hints: consider using nested for loops to write your program. The basic idea is: for each number in array `b`, check whether it is the same as some number in array `a`.

—— YOU CAN STOP HERE OR PROCEED TO GET THE BONUS MARKS ——
6 Bonus Question [25 points]

Write a function to check whether an integer sequence is embedded in another. If a sequence of numbers are “dispersed” in another array in the same order, we say that the sequence is embedded in the second one. For example {1 2 3} is embedded in {1 2 9 8 6 3 5} but {2 1} and {9 1 5} are not. Write a recursive function to solve the problem. The interface of the function is

```java
public static boolean isEmbedded(int[] a, int[] b)
```

[Hint: you may need a helper function `public static boolean isEmbedded(int[] a, int[] b, int al, int ar, int bl, int br)` that returns true if sub-sequence `{a[al]...a[ar]}` is embedded in sub-sequence `{b[bl]...b[br]}`]