University of Illinois at Urbana-Champaign
Department of Computer Science

First Examination

CS 125 Introduction to Computer Science
75 minutes permitted

First, Last name: _____ _____

NetID: _ _ _ _ _ _ _ _ _ @ illinois.edu *(please write legibly)*

Discussion Section (Day/Time or AY_):

• This is a closed book and closed notes exam. No electronic aids are allowed. You must turn this exam booklet in before leaving

• You should have 9 sheets total including one scratch sheet The last sheet is scratch paper; you may detach it while taking the exam, but must turn it in with the exam when you leave.

• You should not write any additional methods for any problem in this exam; you only complete the main() method for that problem. Also, you are not allowed to use the break, continue,or switch statements on this exam.

• Unless we say otherwise in the specific problem, you can assume all values entered by the user will be acceptable input for that program.

• When you write code, you may use a shorthand for System.out and TextIO input and output methods provided it is obvious to the graders which method you are using. For example it is acceptable to use Sopln in place of System.out.println and to use Sopt in place of System.out.print Likewise, you can use T.rlnI(), T.rlnC(), and T.rlnD() in place of TextIO.readlnInt(), TextIO.readlnChar(), and TextIO.readlnDouble().

• For full marks correct syntax is required: Ensure all statements include a semicolon and the correct use of upper/lower case, single quotes and double quotes. However a syntactically correct but flawed program will earn a low score i.e. semantics and correctness is more important than minor typos and syntax errors.

Problem	Points	Score	Grader
1	12		
2	14		
3	10		
4	15		
5	12		
6	12		
Total	75		

CS125 Midterm Exam Spring 2009

1. Type Analysis – 12 points (3 points each)
For each of the code snippets below, there is **zero**, **one** or **more** type errors that the compiler would catch. Explain what the type error is, by indicating what types and operators are involved in the erroneous part of the expression. For example, if the code were `int x = 4 + "hi";` , then the error is "a variable of type int is being assigned the value of an expression of type String". In other words, you need to be specific about what types are involved; you can't just say "this assignment is illegal" in response to the above code. If there is no type error write "No T.E."

(a)
```
int t = TextIO.getlnInt();
char w = TextIO.getlnChar();
boolean invalid = ((t < 1) || (t >= 28.0) && w);
```

(b)
```
double offset = (16 + 1) / 2.0;
int average = (offset + 14.0 + 7) / 3;
```

(c)
```
String text = TextIO.getln();
text  = text.toUpperCase();
int dist = text.length() - text.indexOf("M");
TextIO.putln("The result is " + dist);
```

(d)
```
char[] quote = new char[]{'C','S','1','2','5'};
TextIO.putln(quote[0] - 'A');
```

2. Conditionals – 14 Points

Late Nite Food Inc, recommends food based on the user's preferences. Complete the program below to print out either "Pasta" "Meat" "Chocolate" or "Samples", using the following rules:

Carnivores eat Meat provided they have $8 or greater to spend.
Nibblers eat Chocolate provided they have more than $3 to spend.
Nibblers eat Samples if they have zero dollars to spend.
All others eat Pasta.

You should add if-else statements and **four** output statements to the code below – one to print each output. That is, **do not** print out "P" in three different places in the code – your boolean expressions should be such that you do not have multiple statements in your code which print out the same food item. You may shorten your output to a single letter: "P" "M" "C" "S" (Pasta, Meat, Chocolate, Samples respectively).

```java
public class LateNiteFood {
  public static void main (String[] args){
  TextIO.putln("Are you a Carnivore, Nibbler or Vegetarian?");
  TextIO.putln("Type C N or V and press return");
  char type = TextIO.getlnChar();  // assume either C N V
  TextIO.putln("How many dollars do you have to spend?");
  int dollars = TextIO.getlnInt();  // assume a valid integer
```

```java
  } // end of main method
} // end of class
```

3. Loop Analysis – 10 Points

For this question, consider the following program.

```
public class Mystery {
  public static void main(String[] args) {
    int t = 14;
    int x = 0;
    int iteration = 0;

    TextIO.putln(iteration + " " + t + " " + x); // Line A
    TextIO.putln("Space="+(int)' ');

    while(t>0) {
      x = x + (t % 2);
      t = t/2;
      iteration ++;
      TextIO.putln(iteration + " " + t + " " + x);
    }
  }
}
```

1. Write the Mystery program's output in the space below. The first two lines are completed for you.

```
    0   14   0
    Space=32
```

2. What is the new program's first output line when Line A is replaced with the following?

```
  TextIO.putln(iteration + ' ' + t + ' ' + x); // New Line A
```

New output:

3. What is the new program's first output line when Line A is replaced with the following?

```
  TextIO.put(iteration + " " + t + " " + x + "\n"); // New Line A
```

New output:

4. True or False? The last value of x equals the number of ones in the base 2 representation of 14?

Your Answer:

4. Text Processing – 15 Points

Complete the following program to process the first 99 lines[1] of the Shawshank
Redemption movie script. Your program will count the number of these lines that contain
the word "ANDY" in any mix of uppercase or lowercase. After processing is complete,
print the result in the following exact format:

```
FOUND=10
ABSENT=89
```

Where *10* is replaced by the number of lines containing ANDY in any case and *89* is
replaced by the number of lines that do not. You can assume the file is longer than 99
lines.

```java
public class Shawshank {
  public static void main(String[] args) {
    TextIO.readFile("ShawshankRedemptionScript.txt");




  } // end of main method
} // end of class
```

---

[1] A regular line in a file; i.e. TextIO.getln() will return one line of script.

5. Arrays – 12 Points

```
 public class VotingMachine {
  public static void main(String[] args) {
    TextIO.putln("Number of voters?");
    int len = TextIO.getlnInt();
    int[] votes;
// Space 1:


    for(int i=0; _____ ; _____) { // Line A
      TextIO.putln("Voter "+i+": Candidate 1,2,or 3?");
      votes[i]= TextIO.getlnInt(); // Assume 1 2 or 3
      TextIO.putln("You choice:" + votes[i]);
// Space 2:



    }
    // Print Candidate 3's result first:
    for(int key=3; key>0; key--) {              // Line B
      int count = 0;                            // Line C
      for(int i=0; i<len;i++)
        if(array[i] == key) {
          count++;
        }
      TextIO.putln(key +":" + count);
    } // for
 } // main
} // class
```

1. Write code in space 1 to initialize the *votes* array to hold *len* integers.

2. Briefly describe the purpose of the code between Line B and the end of the program.

3. Complete Line A to read in all *len* voter choices.

4. Describe carefully the effect of swapping the order of Line B and Line C. Which candidate (1,2, or 3) could gain the most reported votes from this alteration?

5. Add code in "Space 2" area to modify just 1% of the voting array. If the 100th, 200th, 300th…etc. voter chooses candidate two, change their vote to be candidate one. In all other scenarios, do not change the voter's choice.

6 (Last Question). Cipher – 12 Points

Complete the following encryption program. The program requests two strings from the user: A lookup pad, exactly 26 characters long, and the message to be encrypted. The program then prints the encrypted version of the message. The encrypted message is created as follows: The letter A is replaced by the first lookup character, letter B is replaced by the second lookup character and so on. The message contains only uppercase letters A to Z, and contains no spaces, punctuation, digits or other characters. Two examples: if the lookup pad is "ABCDEFGHIJKLMNOPQRSTUVWXYZ" then the encrypted message is identical to the original. If the pad is "ZYXWVUT…" then message "ACEE" is encrypted as "ZXVV"

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

    TextIO.putln("Lookup pad (Exactly 26 characters)?");
    String lookup = TextIO.getln();

    TextIO.putln("Message to encrypt (Only A-Z characters)?");
    String mesg = TextIO.getln();
```

Empty Page

Scratch paper