CSci A201/A597

Midterm 1, Spring 1999

2/17/99

Name: _____

Username: _____

Circle the name of your AI (teacher)

Sara Yuming Chris Robert Haiyan Steve

Instructions:

- This is a open book exam. You may use any book, notes, or printouts of web pages you want.
- You may use a calculator if necessary (although I doubt it will be). But the use of laptop computers is not permitted.
- If you have any questions during the exam, please feel free to ask.
- There are total 14 questions on this exam. 10 multiple choice, and 4 short-answer type. All multiple-choice questions carry 3 points, and all short-answer type questions carry 5 points.
- You have exactly two hours to complete this exam.

Page	Points	Score
2	15	
3	15	
4	10	
5	5	
6	5	
Total	50	

For the following multiple-choice questions, there can only be one answer unless otherwise mentioned. Indicate the correct answer by circling the letter corresponding to the answer.

1. Which of the following is not a valid variable name?

   (a) `scndChar`

   (b) `float` **correct** - float is a reserved word for a data type

   (c) `last_item`

   (d) None - all are valid variable names.

2. What is the result of the expression `9*3-5*3/2`?

   (a) 33

   (b) 20 **correct** - precedence and integer division makes 27-7

   (c) -27

   (d) 38

3. Which of the following is a character constant?

   (a) `'x'`

   (b) `'\n'`

   (c) `'\\'`

   (d) All of the above **correct** first one is simple, the others are escape characters

4. What kind of error is in the following line?
   ```
   String hi =  "hello".substring(0,7);
   ```

   (a) Syntax error

   (b) Logical error

   (c) Runtime error **correct** string does not have 7 letters

   (d) No error

5. What does the following code fragment print?
   ```
   System.out.println("You can reach the CS Dept at " + (812)-855-6486);
   ```

   (a) `You can reach the CS Dept at (812)-855-6486`

   (b) `You can reach the CS Dept at 8128556486`

   (c) `You can reach the CS Dept at -6529`

   (d) The program generates an error. **correct** can't subtract number from a string

6. What will the following code fragment print?

```
Time sometime = new Time(1999,2,17,10,25,0);
Time weirdtime = sometime;
if (sometime == weirdtime) { System.out.print("Right time!"); }
else { System.out.println("Oops!"); }
```

   (a) `Right time!` **correct** comparing references

   (b) `Ooops!`

   (c) `Right time!Oops!`

   (d) The code contains an error.

7. What will the following code fragment print?
`System.out.println("Result is " + "Phone".substring(1,3));`

   (a) `Result is Phon`

   (b) `Result is hon`

   (c) `Result is ho` **correct** count.

   (d) The code contains an error.

8. Which of the following expressions will result in a word by capitalizing only the first character and leaving the rest unchanged, given the length of the original word in a variable `len`? This is only an expression, so ignore missing semicolons.

   (a) `word.subString(0,1).toUpperCase() + word`

   (b) `word.charAt(0).toUpperCase() + word.substring(0,len)`

   (c) `word.charAt(0).toUpperCase() + word.substring(1,len)`

   (d) `word.substring(0,1).toUpperCase() + word.substring(1,len)` **correct** toUpperCase is a string method

9. What will be the value of score after performing the following?
```
int score = 20;
score+= score/8 + 3;
```

   (a) 20

   (b) 25 **correct** Assignment to 20 + 5

   (c) 5

   (d) 7

10. Which of the following logical expressions is equivalent to `!(!(x && !y))`?

   (a) `(x && !y)` **correct** double negative is positive

   (b) `(x || !y)`

   (c) `(!x || y)`

   (d) `(!x && y)`

11. (5 points) Given the following declarations:

```
int salary = 25;
String star = "Clint Eastwood";
String movie = "Unforgiven";
```

For each expression in the following table, indicate its value and the type of the value.

Expression	Value	Type
`salary / 2 * 2`	`24`	`int`
`salary % 10`	`5`	`int`
`movie.substring(4, 7)`	`"rgi"`	`String`
`salary + star.substring(0, 5).toUpperCase()`	`"25CLINT"`	`String`
`(10 <= salary) && (salary != 25)`	`false`	`boolean`

12. (5 points) The capital of Venezuela is Caracas. Insert the appropriate boolean expression in the `if` statement below, so that the code properly tests whether or not the user knows the capital of Venezuela. It should not matter if the user mixes upper and lower case letters in their answer.

```
System.out.print("What is the capital of Venezuela? ");
String capital;
capital = Console.in.readWord();

if ( capital.toUpperCase().equals("CARCAS") )
System.out.println("You are correct!");
else
System.out.println("Sorry, but thanks for playing.");
```

13. (5 points) You are an A201 AI and you are holding your office hours. Several bewildered students drop by and show you parts of programs they have written which contain errors that either prevent the program from running, or produce the wrong answer. In each of the following program fragments, correct the student's error by rewriting or inserting one or more lines. You cannot remove any lines. Correct the code so that it matches the expected output shown. **Make as little change to the given code as possible.** You may write any changed line or added line in the space provided, and draw an arrow to where in the code that line should go to.

(a)
```
String weather = "snow";
 weather =  weather + "ing";
System.out.println("Today it is " + weather + " outside.");
```

Expected output: `Today it is snowing outside.`

(b)
```
System.out.print("Enter a word:  ");
String word = Console.in.readWord(); /* suppose user says hello */
int len = word.length();
String first = word.substring(0,1);
String rest = word.substring(0,length);  change to (1, len)
 first =  first.toUpperCase();
System.out.println("Your word with first letter capitalized is " +
first + rest);
```

Expected output: `Your word with the first letter capitalized is Hello`

(c)
```
int x = 5;
x = x / 2.0;
double x = 7.0;  change to x = 7;
System.out.println(x);
```

Expected output: `7`

(d)
```
String vegetable = "carrot";
 String bigVeggie = vegetable.toUpperCase();
System.out.println("bigVeggie = " + bigVeggie);
```

Expected output: `bigVeggie = CARROT`

(e)
```
String fruit = "apple";
int len = fruit.length();
fruit = fruit.substring(1, len).substring(1, len-1);
System.out.println("The wierd answer is " + fruit);
```

Expected output: `The wierd answer is ple`

14. (5 points) **Solving the Year 2000 problem** In this example, you want to solve the Y2K problem using the popular "cutoff year" method. In this method, if the 2 digit year is ¡strong¿below¡/strong¿ the cutoff-limit (*say 20*) then it is treated as after year 2000, and if the 2 digit year is equal to or above the limit, it is treated as before year 2000. You want to write a program that asks a user for a month, day and year, and prints out the Y2K-safe date. Note that the user may type in a Y2K-safe date herself, in which case the program should not change the date. A few runs of this will look like:

```
Enter a date in "month date year" format: 12 12 92
The Y2K safe date is 12/12/1992.
...
Enter a date in "month date year" format: 12 12 1999
The Y2K safe date is 12/12/1999.
...
Enter a date in "month date year" format: 12 12 18
The Y2K safe date is 12/12/2018.
...
```

Write the code for the above program. Some of the code is already written for you. Complete the code by writing in the space provided. **You should not declare any variables other than those already specified.**

```
class Y2Kconvert {
   public static void main(String [] args) {
      int month, date, year;
      final int cutoffYear = 20;

      System.out.print("Enter a date in \"month date year\" format: ");
      month = Console.in.readInt();
      date  = Console.in.readInt();
      year = Console.in.readInt();

      if (year < cutoffYear)
          year += 2000;
      else if (year < 100)
          year += 1900;

      System.out.println("The Y2K safe date is " + month + "/" + date
                           +"/" + year + ".");

   }
}
```