THE UNIVERSITY COLLEGE OF THE CARIBOO

COMPUTING 113

Review Questions

1. *What is the exact from the tricky code segment below?*

```
int a=-10, b=20, c;

c = a/b;
System.out.println ( c );

b++;
c = a + ++b;
a = --a + a++;

System.out.println (a+", "+b);
```

2. *The following loop segments compile and run, but both have errors. Circle the problems for each and fix them!*

a)
```
int sum=0;

  // loop and sum all the values from 1 to 10
for (int i=0; i<10; i++)
{
   sum = sum + i;
   i++;
}
```

b)
```
int input=0, total=0;

while (input != -99)      // loop until user enter -99
{
   System.out.print ("Enter the next value (stop at -99):");
   input = SavitchIn.readLineInt();  // get user value
   total = total + input;            // add value to total
}

System.out.println ("The total is: "+total);
```

3. *Write a static method **drawAngleString()** that has a single parameter **size** and returns a String reference containing an "angle" composed of astericks (stars). There is no maximum size, but negative sizes should be assumed to be a size of zero.*

   For example, assume a size of 4 is passed to the method. It will return a String containing:

```
**
****
******
********
```

4. *Write a static method **swap()** that swaps the elements of an array. The method returns nothing, but has three parameters: an array of type double, and two **int**s for the indexes of the elements to swap.*

5. *Using either **for**, **while**, or **do_while**. Write the code segment (not the whole main()) that asks the user for the number of values that will be be provided (**n**). It then repeats (iterates) for **n**-times, asking the user for a value each time.*
   At the end, the program displays: the maximum value entered and the minimum value entered.

6. *Included in the same program class as the main() that uses it, write a method called **power()**, that returns the value of x^y (x raised to the power of y); assume x is an integer, and y is an always positive integer.*

   Determine the return type, the parameter list, and write the body of the based on the following example of the method call in main():

```
int result=0, a=5, b=3;

result = power(a,b);        // return a raises to the power b
```

7. *Write the complete program class to accomplish the following:*
   *The program accepts a phrase from the user and stores it in a String. From this String, the program loops through the length of the string, and displays a series of *'s on a line equal to the ASCII/Unicode integer value of the character being examined (see example below).*
   *You can use the **.charAt()**,and **.length()** String methods, and don't forget that character positions in strings begin at 0 upto {length of string}-1.*

   Example user dialog:

```
Enter a string: AB C

A: *...*************      displays 65 *'s ; Unicode 65 = 'A'
B: *...**************     displays 66 *'s ; Unicode 66 = 'B'
 : *...*****              displays 32 *'s ; Unicode 32 = ' '
C: *...***************    displays 67 *'s ; Unicode 67 = 'C'
```

   (Hint: The integer value of a character is obtained by using casting*.)*

8. *Write a class **Date**, that contains details of any date: year, month, and day (all as integers).*
   Although simplified, the class ensure only that the month # is 1 to 12 (it does not valid the correct number of days per month).
   Two constructors are required: default that sets all data to 1's, and a full constructor for all data (ensuring the correct number of months; set to all 1's if incorrect).
   Other necessary methods are:
   - *accessors to return each individual attribute: getYear(), getMonth(), getDay()*
   - *one mutator that changes all attributes, while ensuring a valid year*
   - *the .toString() method that returns a String with the date in the format: YY / MM / DD*
   - *an .equals() method that returns of the current object is the same as another object (year, month, and day must match)*

9. *Write a class **SportsEventInfo**, that describes the specifics of general sporting activities, used in a database for the Amateur Sporting Events catalogue in the city.*

   This class requires the following attributes (instance variables):
   - *name of event*
   - *name of organising group*
   - *maximum number of participants*
   - *date (an object of the Date class)*
   - *a boolean flag for whether the event is for charity, or not.*

   This necessary methods:
   - *default constructor (sets all to "zero" or default values)*
   - *full constructor setting all (the date is given as three integers (year, month, day) and an object must be created before being assigned to the instance variable)*

   - *mutator to change all the attributes by asking the user*
   - *mutator to change the number of participants (ensuring a non-negative value)*
   - *mutator to change the status of the charity flag*
   - *accessor to return the name of the event*
   - *accessor to return the status of the charity flag*
   - *.equals() method to determine if the current object equals another object, but only checking the name (event and organising group) and date*
   - *.toString() that returns a String containing the object's data*

   *Write the statements in a **main()**, that declares an object of this class (setting the data as below), and performs the necessary operations,*
   - *new object event: "Kids Group Baseball", "The After Schools Club", 200, (2005/04/22), No*
   - *display the name of the event and the group to the user*
   - *change the charity status so that it is a charity event*

   - *declare a new object using the default constructor, then the mutator to get all the data from the user*

   - *examine if the two objects are equal using an "if" (display the appropriate message depending if they are equal or not).*