CSC 1051 – Algorithms and Data Structures

Midterm Examination - Sample algorithm questions.

Construct an algorithm that inputs a number num and prints all its factors (i.e., all the values of \( x \) in the range \( 2 \ldots (\text{num}-1) \) that such that \( \text{num} \) is divisible by \( x \)). After the factors are printed, print a goodbye message.

**Example:** If \( \text{num} \) (i.e., the input) is 20, the algorithm should print:

```
The factors of 20 are:
2  4  5  10
Goodbye
```

**Directions:**
Write your algorithm by rearranging and structuring elements chosen from the list below, using indentation to show structure. Do not use anything else and note that not all of these are needed, but you may use one of them more than once, if necessary.

- `input num`
- `input x`
- `x = 1`
- `x = 2`
- `x = x + 1`
- `if (num \% x == 0)`
- `if (num \% 2 == 0)`
- `else`
- `while (x < num)`
- `while (x <= num)`
- `while (num \% 2 == 0)`
- `print ''The factors of '' num '' are:'''`
- `print x`
- `print num`
- `print ''Goodbye''`

---

Construct an algorithm that inputs some numbers from the user, terminated with a zero (sentinel value). The algorithm should print the numbers entered and compute and print their sum. After the sum is printed, print a goodbye message.

**Example:** If the numbers 5 -3 8 0 are entered as input, the algorithm should print:

```
5
-3
8
Sum = 10
Goodbye
```

[Note that the terminating zero should NOT be printed.]

**Directions:**
Write your algorithm by rearranging and structuring elements chosen from the list below, using indentation to show structure. Do not use anything else and note that not all of these are needed, but you may use one of them more than once, if necessary.

- `input num`
- `input sum`
- `num = 0`
- `sum = 0`
- `num = num + 1`
- `sum = sum + 1`
- `sum = sum + num`
- `sum = num + 1`
- `if (sum == num)`
- `if (sum < num)`
- `if (num < sum)`
- `else`
- `while (num < sum)`
- `while (sum < num)`
- `while (num != 0)`
- `while (num != sum)`
- `print ''The factors of '' num '' are:'''`
- `print ''Sum = '' sum`
- `print num`
- `print ''Goodbye''`
Midterm Examination - Sample algorithm questions

Construct an algorithm that inputs an integer num and 10,000 other integers and prints a message indicating whether num was found among the 10,000 other integers, followed by a goodbye message.

Hint: You need to use a boolean variable found to keep track of whether you found a match.

Example: If num (i.e., the first number input) is 1318, and 10,000 other numbers are input after that, none of which is equal to 1318, the algorithm should print:

```
Searching for 1318
Not found
Goodbye
```

Alternatively, if the number 1318 occurred one or more times among the other numbers, it should print:

```
Searching for 1318
Found it!
Goodbye
```

Directions:
Write your algorithm by rearranging and structuring elements chosen from the list below, using indentation to show structure. Do not use anything else and note that not all of these are needed, but you may use one of them more than once, if necessary.

```
found = true
found = false
input num
input x
count = 1
count = count + 1
x = x + 1
if (x == num)
if (found)
if (found == true)
if (found == false)

if (x != num)
else
while (count <= 10000)
while (num <= 10000)
while (count <= num)
print "Searching for “ num
print x
print num
print "Found it!"
print "Not found"
print "Goodbye"
```

Construct an algorithm that inputs a number num and then generates and prints num random integers in the range 0…9. After the random numbers are printed, print a goodbye message.

Example: If num (i.e., the input) is 5, the algorithm should print something like this:

```
5 random numbers:
8 4 8 0 3
Goodbye
```

Directions:
Write your algorithm by rearranging and structuring elements chosen from the list below, using indentation to show structure. Do not use anything else and note that not all of these are needed, but you may use one of them more than once, if necessary.

```
input num
input count
input x
x = random(10)
x = x + 1
count = 1
count = count + 1
num = num + 1
if (x < num)
```

```
if (x <= num)
else
while (count <= num)
while (x <= num)
while (count <= x)
print x
print num " Random numbers: "
print "Goodbye"
```
1. Suppose you look up a method in the Java API and find this method heading:

   ```
   String doThis ( int x )
   ```

   a) Fill in the following information about this method:

   Method name: ________________
   Method return type: __________
   Required parameters for the method: (how many? of what type(s)?)
   _______________________________________________________

   b) Suppose this method is defined in class QuizClass and you have an object quizzie of that class. Give an example of the use of the dot operator to invoke this method through quizzie to print something. (Hint: You need to use quizzie or QuizClass, but not both.)

2. Suppose the String variable message is already initialized. Write a Java code fragment that prints out message, one letter per line.

3. Write a declaration for a String variable called eeeful and initialize it to the characters stored in another String object called original with all 'a' characters changed to 'e'.

4. a) Write code to declare and instantiate an object of the Random class (call the object reference variable rand).
   
   b) Write an expression using the nextInt method that generates random numbers in the range 1 to 1000, including the endpoints. Use the version of the nextInt method that accepts a single integer parameter.
   
   c) Repeat using Math.random()

5. Write some code that uses the Math class to compute and print out the value of the square root of the sine of the angle 1.25.