# New York University
# Department of Computer Science

# CS101-004– Mock Midterm 1
# Intro to Computer Science

Instructor: Tobias Blickhan

-

**Name**: _____
**Net ID**: _____

---

This is a mock exam. It does not count towards your grade.

There are two extra problems at the end for you to practice with. The following instructions are as you get them in the real exam:

- Fill in your full name and NetID on this cover page.
- This exam contains 9 pages (including this cover page) and 5 questions. When the exam starts, make sure you are not missing any sheets.
- Total of points is 50. You have 50 minutes to answer all questions. This corresponds to roughly one point per minute. Manage your time.
- There are two blank pages at the end of the exam that can be used as scrap paper.
- If you have any questions during the exam or need more paper, raise your hand and I will get to you.
- It needs to be clear what question your answers correspond to. If it gets messy or if you write something on the scrap paper that should get graded, clearly indicate it. If you do not want something graded, cross it out. In the case where you give two answers to a question, and one is incorrect, you get no points.
- If necessary, comment your code so that it clear what you want to do. You do not get points for pseudo-code or comments, but you do get points for correct code that follows them.
- This exam is closed books and no electronic devices are allowed. You are allowed one page of notes.

1. (15 points) Short questions (multiple choice/fill in the blank type). Similar to what you know from the quizzes. Skipped here.

2. (15 points) Write a method that determines if a given string is a palindrome, i.e. if it is identical when read backwards such as "hannah" or "step on no pets". For the purpose of this problem, capitalization, spaces, and punctuation matter, so "Step on no pets" is not a palindrome and neither is "was it a car or a cat i saw". The method should return true if the input is a palindrome.

   You can (and should) use the following method:

```
char java.lang.String.charAt(int index)
    Returns the char value at the specified index.
    An index ranges from 0 to length() - 1.
    The first char value of the sequence is at index 0,
    the next at index 1, and so on, as for array indexing.
Parameters:
    index the index of the char value.
Returns:
    the char value at the specified index of this string.
    The first char value is at index 0.
Examples:
    "cat".charAt(1) // 'a'
    "shark".charAt(4) // 'k'
```

   Do not use tools from StringBuilder such as reverse(). You have to implement the logic of the method yourself.

(continued on next page)

This method takes as input a String and returns a boolean. Complete and use the following method header:

```
public static ..... isPalindrome( ..................... )
```

3. In this question, you will write a method that takes as input a strictly positive integer and checks if it is a prime number. '1' is not a prime number.

   (a) (5 points) Draw the activity diagram of such a method. You do not need to check input validity.

(b) (15 points) Implement a method that checks if a given strictly positive integer is a prime. It takes an int and returns a boolean. Recall that you can use the modulo operator % to check if one integer is divisible by another. You do not need to check input validity. Complete and use the following method header:

```
public static ..... isPrime( ..................... )
```

4. (0 points) (Extra question for exam prep) Write a method that, given an array a of doubles, returns a "smoother" array by taking averages as follows:

$$\mathtt{a}[i]_{\text{out}} = \frac{1}{4}\mathtt{a}[i-1]_{\text{in}} + \frac{1}{2}\mathtt{a}[i]_{\text{in}} + \frac{1}{4}\mathtt{a}[i+1]_{\text{in}} \tag{1}$$

In this formula, when $i = 0$, $\mathtt{a}[i-1]$ (which would normally cause an error) should be the last element of the array. Likewise, if $i$ is the last index, $\mathtt{a}[i+1]$ should be the first element. Do not modify the input data.

Complete and use the following method header:

```
public static ..... smooth( ..... )
```

5. (0 points) (Extra question for exam prep, hard to do on paper!) In this question, we design a pseudo-random number generator. We use the "middle square method", proposed by (the genius) John von Neumann in 1949. The method is given two inputs, a "seed" (a 4-digit integer), and the amount of random numbers to be generated. To generate the next random number, we square the input and write the result as a 8-digit integer. If the squared input does not have 8 digits, we pad it with zeros from the start. The next random number is then given by the middle 4 integers in the resulting number. This number is printed and then used used as input for the next iteration.

   For illustration, here is the whole workflow on an example:

   1. We start with a seed, say 1234
   2. Square it, giving 1522756
   3. Pad with zeros, giving 01522756
   4. Take the middle four digits, giving 5227
   5. Print 5227 and use it as new input
   6. Square it, giving 27321529
   7. Padding not needed
   8. Take the middle four digits, giving 3215
   9. Print 3215 and use it as new input
   10. Square it, giving 10336225
   11. Padding not needed
   12. Take the middle four digits, giving 3362
   13. Print 3362 and use it as new input
   14. ...

   Implement the random number generator: pad should take an int and return a String. middleDigits should take a String and return an int.
   Useful functions for this part: substring, String.valueOf, Integer.parseInt.

This page is intentionally left blank.

This page is intentionally left blank.