

First Name: _____ Last Name: _____

McGill ID: _____ Section: _____

Faculty of Science
COMP-202A - Foundations of Computing (Practice) - All Sections
Midterm Examination

Examiners: Practice

Instructions:

• DO NOT TURN THIS PAGE UNTIL INSTRUCTED

- This is a **closed book** examination; only a letter-sized (8.5" by 11") **crib sheet** is permitted. This crib sheet can be single or double-sided; it can be handwritten or typed. Non-electronic translation dictionaries are permitted, but instructors and invigilators reserve the right to inspect them at any time during the examination.
- Besides the above, only writing implements (pens, pencils, erasers, pencil sharpeners, etc.) are allowed. The possession of any other tools or devices is prohibited.
- Answer **all** questions on the scantron sheet.
- This examination has **13** pages including this cover page, and is printed on both sides of the paper. On page 13, you will find information about **useful classes and methods**. **You may detach page 13 from the examination if you wish.**
- **MAKE SURE TO WRITE YOUR NAME AND STUDENT ID ON THE SCANTRON AS WELL AS TO FILL IN THE BUBBLE PROPERLY AT THE BEGINNING OF THE EXAM**

Scoring

The exam will be scored as follows:

1. **This is a practice mid-term examination**

Version-1

Regular multiple choice: 2 points each

1. Which of the following is NOT a legal Java statement:

- (A) `double x = 3;`
- (B) `double x = 3.0;`
- (C) `int x = 3;`
- (D) `double x = "3";`
- (E) `double x = 7 / 2;`

2. How many times is the condition of the while loop checked in the following code?

```
int i = 0;
while (i < 10) {
    System.out.println("Repeat!");
    i++;
}
```

- (A) 0
- (B) 9
- (C) 1
- (D) 11
- (E) 10

3. What will print in the following code snippet?

```
int i = 1;
int j = 2;
i = j;
j = i;
System.out.println(i + " " + j);
```

- (A) 3
- (B) 2 2
- (C) 1 2
- (D) 2 1
- (E) 1 1

4. Suppose you have a method with the following header:

```
public static void fun(double x, int y)
```

Which of the following are legal ways to call the method:

- I. `fun(3, 4.0);`
- II. `fun(3, 4);`
- III. `fun(4.0, 3);`

- (A) II only
- (B) I, II, and III

- (C) I only
(D) III only
(E) II and III
5. Which of the following Java expressions do NOT have a value of 0.
- (A) `(int)((double)1) / 2`
(B) `1 / 2`
(C) `(double)(1 / 2)`
(D) `((double)1) / ((int)2)`
(E) `(int)1.0 / 2`
6. Which of the following Java expressions have a value of 0.
- (A) `1 / 2.0`
(B) `"1.0 / 2.0"`
(C) `1.0 / 2.0`
(D) `(double)(1 * 2)`
(E) `(int)1.0 / 2`

Type of Errors

For each of the following answer whether they would cause an compiler error, runtime error, or a bug. In other cases the mentioned error may not cause a problem for the computer and is merely a question of style. In this case you should choose "Style" In other cases the question may not cause a problem at all in which case you should answer "No Error"

For the purpose of this question, you can use the following definitions:

- **Compiler error** : An error that occurs during the compilation phase of your development. In this case no .class file is produced.
- **Runtime error** : An error that occurs while you are running your program. Your program starts to execute but terminates before the main method has finished executing.
- **Logical error or bug** : An error that results in the wrong results being seen
- **Style error** : The error does not make a difference to the computer but is something that may be considered bad code.

7. Omitting a semi-colon at the end of a variable assignment.
- (A) Compiler Error
(B) Bug (logical) Error
(C) Style
(D) Runtime Error
(E) No Error
8. Declaring two variables with the same name in different methods.
- (A) Style
(B) Compiler Error
(C) Runtime Error

- (D) Bug (logical) Error
- (E) No Error

9. Dividing an int expression by 0.

- (A) Style
- (B) No Error
- (C) Bug (logical) Error
- (D) Compiler Error
- (E) Runtime Error

10. Omitting the main method from a class and then executing that class.

- (A) Compiler Error
- (B) Style
- (C) Runtime Error
- (D) Bug (logical) Error
- (E) No Error

11. Storing a double expression into an int.

- (A) Bug (logical) Error
- (B) Compiler Error
- (C) Style
- (D) Runtime Error
- (E) No Error

12. What prints as a result of the following code being run?

```
public static void main(String[] args) {  
    int x = 8;  
    int y = 3;  
    foo(x,y);  
    System.out.println(x + " " + y);  
}
```

```
public static int foo(int x, int y) {  
    x = x * 2;  
    int temp = x;  
    x = y;  
    y = y * 2;  
    x = temp;  
    return x;  
}
```

- (A) 8 3
- (B) 16 6
- (C) 16 8
- (D) 3 8
- (E) 16 3

13. What prints as a result of the following code being run?

```
public static void main(String[] args) {  
    int x = 8;  
    int y = 3;  
    foo(x,y);  
    foo(x,y);  
}  
  
public static int foo(int y, int x) {  
    System.out.print(y + " " + x + " ");  
    int temp = x;  
    x = y;  
    y = x;  
    return y;  
}
```

- (A) 3 8 8 8
- (B) 8 3 3 8
- (C) 8 3 8 3
- (D) 3 8 8 3
- (E) 3 8 3 8

14. What value is returned by the following method if the input array is { 2, 3, 4, 1, 2, 1}

```
public static int mystery(int[] array) {  
    int foo = 1;  
    for (int i = 0; i < array.length; i++) {  
        foo = foo * array[i];  
    }  
  
    return foo;  
}
```

- (A) 4
- (B) 48
- (C) 2
- (D) 16
- (E) 24

15. What will the following method print when the input number n has value 5

```
public static void mystery(int n) {  
    for (int i = -1; i < n; System.out.print(i + " ")) {  
        i++;  
    }  
}
```

- (A) The code does not compile.
- (B) 0 1 2 3 4
- (C) -1 0 1 2 3 4 5
- (D) -1 0 1 2 3 4
- (E) 0 1 2 3 4 5

16. Suppose `m` and `n` are both `int` variables with positive values. Which of the following boolean expressions are guaranteed to be equal in value to the expression that follows?

`(m % n == 0 && n % m == 0)`

- I. `m == n`
- II. `!(m < n || m > n)`
- III. `!(m < n && m > n)`

- (A) I and II
(B) I, II, and III
(C) I and III
(D) None of the expressions are guaranteed to be the same.
(E) I only

17. How many times does the letter A print in the following code?

```
for (int i = -1; i < 3; i++) {  
    System.out.println("A")  
}
```

- (A) 0
(B) 2
(C) 3
(D) 1
(E) 4

18. How many times does the letter B print in the following code?

```
for (int i = 0; i < 6; i++) {  
    for (int j = 0; j <= 5; j++) {  
        System.out.println("B");  
    }  
}
```

- (A) 36
(B) 0
(C) 30
(D) 5
(E) 11

19. How many times does the letter C print in the following code?

```
for (int i = 0; i < 3; i++) {  
    for (int j = 0; j < 3; j++) {  
        for (int k = 0; k < 2; k++) {  
            System.out.println("C");  
        }  
        System.out.println("C");  
    }  
    System.out.println("C");  
}
```

- (A) 36
- (B) 18
- (C) 30
- (D) 8
- (E) 24

20. How many times does the letter D print in the following code?

```
for (int i = 0; i < 3; i++) {  
    for (int j = i+1; j < 3; j++) {  
        System.out.println("D");  
    }  
}
```

- (A) 1
- (B) 2
- (C) 3
- (D) 6
- (E) 0

21. What prints in the following question.

```
int[][] x = { { 1, 2, 3} , {4, 5}, {6, 7}};  
x[0] = new int[4];  
System.out.println(x.length);
```

- (A) 8
- (B) There is a compiler error.
- (C) 7
- (D) 3
- (E) 2

22. What does the following method return when the input array is { 1, 2, 3 } ?

```
public static int mystery(int[] array) {  
    int i = 0;  
    int count = 0;  
    while (i <= array.length) {  
        if (array[i] % 2 == 0) {  
            count++;  
        }  
  
        i++;  
    }  
  
    return count;  
}
```

- (A) There is a runtime error.
- (B) 0
- (C) There is a compiler error.

- (D) 3
- (E) 2

23. What String does the following code return if the input to the method is the String with contents ABCD.

```
public static String mystery(String input) {  
    String tricky = "";  
    for (int i = 0; i < input.length(); i++) {  
        tricky = input.charAt(input.length() - i - 1) + tricky;  
    }  
  
    return tricky;  
}
```

- (A) ABCD
- (B) DDCCBBAA
- (C) AABBCDD
- (D) DCBA
- (E) There is a runtime error.

24. What String does the following code return if the input to the method is the String with contents ABCDE.

```
public static String mystery(String input) {  
    String tricky = "";  
    for (int i = 0; i < input.length(); i++) {  
        tricky = tricky + input.charAt(i);  
        i++;  
        tricky = tricky + input.charAt(i);  
    }  
  
    return tricky;  
}
```

- (A) AABBCDDEE
- (B) EEDDCCBBAA
- (C) There is a runtime error.
- (D) EDCBA
- (E) ABCDE

25. What String does the following code return if the input to the method is the String with contents ABCD.

```
public static String mystery(String input) {  
    String tricky = "";  
    for (int i = 0; i < input.length(); i++) {  
        tricky = tricky + input.charAt(i);  
        i++;  
        tricky = tricky + input.charAt(i-1);  
        i--;  
    }  
  
    return tricky;  
}
```


- (A) DCBA
- (B) AABBBCCDD
- (C) DDCCBBAA
- (D) There is a runtime error.
- (E) ABCD

Longer multiple choice questions (3 points each)

Coding Question - Fibonacci sequence

You are to write a method named “Fibo” that takes as input an integer n , and returns the n^{th} number in the Fibonacci sequence. If $\text{Fibo}(n)$ denotes the n^{th} number in the Fibonacci sequence, then recall for $n > 2$, $\text{Fibo}(n) = \text{Fibo}(n-1) + \text{Fibo}(n-2)$, and $\text{Fibo}(1) = 1$ and $\text{Fibo}(2) = 1$. You are to assume that the user is always going to enter an input greater than 2. For example $\text{Fibo}(4) = 3$.

Use the 7 following questions to “write” your code. Pay close attention to how your answers group together with each other.

26. Line 1

- (A) `public static void main(String[] args) {`
- (B) `public static int Fibo(int n) {`
- (C) `public static void Fibo(Double input) {`
- (D) `public static int Fibo() {`
- (E) `public static void Fibo(int n) {`

27. Line 2

- (A) `n +=1;`
- (B) `int fib1 = 1; int fib2 = 1;`
- (C) `int[] = new int[n];`
- (D) `int n = 100; n-=0;`
- (E) None of these statements are useful when paired with other questions, leave blank.

28. Line 3

- (A) `int temp = fib1+fib2;`
- (B) None of these statements are useful when paired with other questions, leave blank.
- (C) `while(fib1< temp) {`
- (D) `int temp, counter;{`
- (E) `int temp = 0; int counter = 2;`

29. Line 4

- (A) `while(counter<n){`
- (B) `}`
- (C) `while(true) {`
- (D) `return Fibo(n-1);`
- (E) `temp = fib1 + fib2;`

30. Line 5

- (A) counter ++;
- (B) temp = temp+fib1; fib2=fib1; fib1 = temp;
- (C) return temp;
- (D) return fib1 + fib2;
- (E) temp = fib1+fib2; fib1 = fib2; fib2 = temp;

31. Line 6

- (A) temp++;}
- (B) None of these statements are useful when paired with other questions, leave blank.
- (C) System.out.print(temp);}
- (D) counter++; }
- (E) fib1 = fib2 + temp; }

32. Line 7

- (A) return temp;}
- (B) return counter;}
- (C) } }
- (D) return ++temp;}
- (E) None of these statements are useful when paired with other questions, leave blank.

Coding Question - Sorted array

You are to write a method that takes as input an array of integers, returns **true** if the array is sorted, and returns **false** otherwise. For example, if *a* is the array {1,12,5,6,2,87}, then the method **has** to return **false** as it is not sorted.

Use the 5 following questions to “write” your code. Pay close attention to how your answers group together with each other.

33. Line 1

- (A) public static void Sort(int[] a) {
- (B) public static int[] Sorted(int a) {
- (C) public static boolean Sorted(int[] a) {
- (D) public static boolean Sorted() {
- (E) public static void main(String[] s) {

34. Line 2

- (A) for (int i = -1; i<a.length;i++){
- (B) for (int i = 0; i<a.length-1;i++){
- (C) for (int i = 0; i<a.length;i++){
- (D) int b = a[0];
- (E) for (int i = 0; i<a.length-1;i++){

35. Line 3

- (A) if (a[i] > a[i+1]) { return false; }

- (B) `if (b > a[i+1]) { return b; }`
- (C) `if (a[i] > a[i-1]) { return true; }`
- (D) `if (a[i] > a[i+1]) { return true; }`
- (E) `i++;`

36. Line 4

- (A) `return b; }`
- (B) `return false;`
- (C) `i++;`
- (D) `return true; }`
- (E) `}`

37. Line 5

- (A) `i++; }`
- (B) `System.out.print(a); }`
- (C) `return true; }`
- (D) `}`
- (E) `return false; }`

True false questions (1 point each)

In the following questions answer A for true or B for false.

38. In Java, methods can be defined inside of other methods.

- (A) TRUE
- (B) FALSE

39. In Java, you can change the value of a reference variable declared in the main method in another method.

- (A) TRUE
- (B) FALSE

40. In Java, all methods must be declared inside of a class.

- (A) TRUE
- (B) FALSE

41. A variable of type `char` can be used to store text such as `HelloWorld!`

- (A) FALSE
- (B) TRUE

42. The Java compiler can detect infinite loops.

- (A) FALSE
- (B) TRUE

43. The Java compiler can detect array out of bounds errors.

- (A) TRUE
- (B) FALSE

44. The Java compiler can detect undeclared variable errors.

- (A) TRUE
- (B) FALSE

45. What is the value of the following boolean expression?

```
true && (false || !false) && (true || false)
```

- (A) FALSE
- (B) TRUE

46. Once an array is created, you can not change its size.

- (A) TRUE
- (B) FALSE

47. The following code prints 5.

```
System.out.println("" + 3 + 2);
```

- (A) FALSE
- (B) TRUE

48. Computer programming is fun all the time! (This question will not be graded)

- (A) TRUE
- (B) FALSE

49. I wish there were written programming questions on the exam! (This question will not be graded)

- (A) FALSE
- (B) TRUE

SUMMARY OF JAVA STANDARD LIBRARY METHODS FOR SELECTED CLASSES

• String (package java.lang) Methods:

- public boolean equals(Object anObject): Compares this String to anObject.
- public int length(): Calculates the length of this String.
- public boolean equalsIgnoreCase(String anotherString): Compares, ignoring case considerations, this String to anotherString.
- public int compareTo(String anotherString): Compares this String to anotherString lexicographically; returns a negative value if this String occurs before anotherString, a positive value if this String occurs after anotherString, and 0 if both Strings are equal.
- public int compareToIgnoreCase(String anotherString): Compares, ignoring case considerations, this String to anotherString lexicographically; returns a negative value if this String occurs before anotherString, a positive value if this String occurs after anotherString, and 0 if both Strings are equal.
- public String substring(int start, int finish): Returns a new String composed of the this String starting from index start and up to, but not including index of finish
- public String replace(char c, char d): Returns a new String with all occurrences of the character c in the this String replaced by the character d.
- public char[] toCharArray(): Converts this String to a new character array.

• File (package java.io) Methods:

- public FileSString pathname(): Creates a new File instance that corresponds to the given pathname.

• Scanner (package java.util) Methods:

- public Scanner(InputStream source): Constructs a new Scanner that produces values scanned from the specified input stream.
- public Scanner(File f): Constructs a new Scanner that produces values scanned from the specified File
- public double nextDouble(): Scans the next token of the input as a double.
- public boolean nextBoolean(): Scans the next token of the input as a boolean.
- public int nextInt(): Scans the next token of the input as an int.
- public String nextLine(): Advances this Scanner past the current line and returns the input read.
- public boolean hasNextLine(): Checks whether there are further lines left to scan.

• PrintStream (package java.io) Methods:

- public void print(boolean b): Prints boolean value b.
- public void print(double d): Prints double value d.
- public void print(int i): Prints int value i.
- public void print(Object o): Prints Object o.
- public void print(String s): Prints String s.
- public void println(): Terminates the current line by writing the line separator string.
- public void println(boolean b): Prints boolean value b and then terminates the line.
- public void println(double d): Prints double value d and then terminates the line.
- public void println(int i): Prints int value i and then terminates the line.
- public void println(Object o): Prints Object o and then terminates the line.
- public void println(String s): Prints String s and then terminates the line.

• Math (package java.lang) Methods:

- public static double pow(double a, double b): Returns the value of a raised to the power of b.
- public static double sqrt(double a): Returns the correctly rounded positive square root of double value a.
- public static double random(): Returns a double value with a positive sign, greater than or equal to 0.0 and less than 1.0.
- public static double exp(double a): Returns Euler's number e raised to the power of double value a. (base e) of double value a. of double value a.