



**NAMIBIA UNIVERSITY  
OF SCIENCE AND TECHNOLOGY**

**FACULTY OF COMPUTING AND INFORMATICS  
DEPARTMENT OF COMPUTER SCIENCE**

<b>QUALIFICATION:</b> Bachelor of Computer Science , Bachelor of Computer Science in Cyber Security, Bachelor of Informatics	
<b>QUALIFICATION CODE:</b> 07BACS, 07BCCS, 07BAIF	<b>LEVEL:</b> 5
<b>COURSE:</b> Programming 1	<b>COURSE CODE:</b> PRG510S
<b>DATE:</b> June 2018	<b>PAPER:</b> THEORY
<b>DURATION:</b> 2 Hours	<b>MARKS:</b> 100

<b>FIRST OPPORTUNITY EXAMINATION</b>	
<b>EXAMINER</b>	MR HERMAN KANDJIMI MR SIMON H. MUCHINENYIKA MS NDINELAGO NASHANDI MR LAMECK AMUGONGO MR JEREMIAH LUMBASI MR EDMORE CHIKOHORA MR MUNYARADZI MARAVANYIKA
<b>MODERATOR:</b>	MR COLIN STANLEY

**THIS EXAMINATION PAPER CONSISTS OF 6 PAGES  
(INCLUDING THIS FRONT PAGE)**

**INSTRUCTIONS**

1. Answer **all** questions.
  2. Total marks per question are given in [].
  3. **Read and understand the question carefully before attempting to answer**
  4. When writing take the following into account: The style should inform than impress, it should be formal, in third person, paragraphs set out according to ideas or issues and the paragraphs flowing in a logical order. Information provided should be brief and accurate.
  5. Please, ensure that your writing is **legible, neat and presentable**.
- PERMISSIBLE MATERIALS**
6. Calculator.

1. Fill-out the Table below.

[10]

Data type	Value	Declarations and assignment
short	25	short smallNumber = 25;
char		
	true	
		float num = 36.2f;
	"My name is Bill"	
	7789.8	

2. Indicate whether the following statements are TRUE or FALSE

[10]

- A) Strings, char and double are all examples of primitive data types in Java.
- B) In Java, the identifiers student, Student, and sTudent are all different.
- C) Every source file must be named the same as the class declared in the file
- D) After an Array is declared the size can easily be changed.
- E) The == operator can be used to compare two String objects.
- F) One array in the Java programming language has the ability to store many different types of values.
- G) For the expression  $(y \geq z \mid \mid a == b)$  to be true, at least one of  $(y \geq z)$  or  $(a == b)$  must be true.
- H) Consider the statement `examAdmission = (score >= 50) ? "Allowed" : "Denied";` then the value of examAdmission is Denied, if score = 49
- I) The statement `if(!allowed)` will evaluate to false when allowed = true
- J) The Java + operator is used for both string concatenation and addition.

3. What is the difference between the following:

- A) Compiler and assembler [3]
- B) Assembly language and Machine language [3]
- C) Primitive and Non-Primitive data types [3]

4. What is the output of the following program and show all your workings?

[6]

```
public class Exam_Q4
{
    public static void main(String[] args)
    {
        System.out.println(mystery(5));
    }
    public static int mystery(int n)
    {
        if (n <= 1)
            return 1;
        else
            return (mystery(n - 1) + n);
    }
}
```

5.

A) Rewrite the following piece of code using a **do-while** loop.

[5]

```
public static void Display(String[] studentNames) {  
    for(int i = 0; i < studentNames.length; i++){  
        System.out.println(studentNames[i]);  
    }  
}
```

B) Rewrite the following code snippet using a switch statement.

[5]

```
if(rating == 'E') //Excellent  
    System.out.println("You must see this movie!");  
else if(rating == 'A') //Average  
    System.out.println("This movie is OK, but not great.");  
else if(rating == 'B') //Bad  
    System.out.println("Skip it!");  
else  
    System.out.println("Something is wrong.");
```

6. Create a method that takes in three arrays, one for student names, one for test marks and a last one for assignment mark. The method should then calculate the students qualifying mark (a student needs 50 or more to qualify) using the following weights: 40% of the test and 60% of the assignment, finally print out whether the person qualified or not. [10]

Use the below format for you print out:

Name	Test	Assignment	Final	Examination
-----	-----	-----	-----	-----
Lovisa	59	85	75	Allowed
John	52	45	48	Denied

7. Examine the code snippet below and answer the questions that follow:

```
4 public static void main(String[] args) {
5     int num1,num2,sum = 0; //variable declartion
6     Scanner kbdInput = new Scanner(System.in);
7
8     System.out.println("Input two whole numbers: ");
9     num1 = kbdInput.nextInt();
10    num2 = kbdInput.nextInt();
11
12    int start = (num1 < num2) ? num1:num2;
13    int end = (num1 > num2) ? num1:num2;
14
15    while(start <= end ){
16        if(start != end)
17            System.out.print(start+" + ");
18        else
19            System.out.print(start+" = ");
20        //calculating the sum
21        sum += start;
22        start++;
23    }
24    System.out.println(sum);
25    //Add code for average below
26
27 }
```

- A) What does line 8 do, and how is this called in Programming? [2]
- B) Explain what happens line 9? [2]
- C) What does line 12 do, and how is this known in Java? [2]
- D) By making use of 9 and 5 as user input, briefly explain what is the purpose of this whole code snippet? [3]
- E) Line 25 provides a comment that requires you to add a new piece of code for calculating the average, provide this requested code.[Hint: the quantity of the numbers is the difference between the num1 and num2, using Math.abs() ] [3]

8. Examine the code snippet below, identify line with errors and correct them. Finally indicate the output of the code snippet.

```
5      char symbol = 'B';
6      //array to store wages for 3 employees
7      double wages[] = new double[ ];
8      String employeeNames[] = {"Anna", "George", "Van Wyk"};
9      String employeeName = 'Trump';
10     employeeNames[1] = employeeName;
11     String workPlace = "Welcome to Wakanda : Software Engineers";
12     workPlace = workPlace.replaceFirst(":", "-");
13     workPlace = workPlace.substring(11);
14     int hours[] = {45,25,18}
15     double rate Per Hour = 105;
16
17     System.out.println(workPlace.toUpperCase());
18     for(int i =0;i < wages.length;i++){
19         wages[i] = (++hours[i]) * ratePerHour;
20         System.out.println(employeeNames[i]+" Salary NS "
21                             + String.format("%.2f", wages[i]) );
22     }
```

- A) Line with errors and Solutions: [4]
- B) Code output : [4]

9. Given an employee's salary from user input, if the salary is even then increment the salary by 5% otherwise increase by it by 4%. Next the salary has to be deducted a tax, according to the following table:

Salary Range	Percent of Tax
Salary $\geq$ 12000	12.5%
8000 $\leq$ Salary $<$ 12000	10%
5000 $\leq$ Salary $<$ 8000	8%
2000 $\leq$ Salary $<$ 5000	5%
Salary $<$ 2000	1%

The system should then print out the current employee's Salary and the percent of tax paid.

- A) Write a Pseudocode to solve the above problem [5]
- B) Create a flowchart for the above pseudocode. [8]
- C) With the help of both your Pseudocode and Flowchart, create a Java program that solves the program as per the given problem description [12]

[END]