

ASSIGNMENT-6

DATE:13/7/24

1. Write a program to given an integer array nums, find the subarray with the largest sum, and return its sum.

Input nums = [-2, 1, -3, 4, -1, 2, 1, -5, 4]

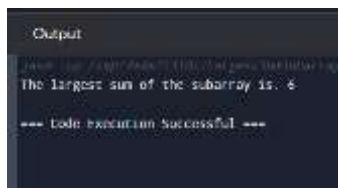
Output: 6

Explanation: The subarray [4,-1, 2, 1] has the largest sum 6

CODE:

```
public class LargestSumSubarray {  
    public static void main(String[] args) {  
        int[] nums = {-2, 1, -3, 4, -1, 2, 1, -5, 4};  
        System.out.println("The largest sum of the subarray is: "+maxSubArray(nums));  
    }  
    public static int maxSubArray(int[] nums) {  
        int maxfir=nums[0];  
  
        int maxlast=nums[0];  
        for (int i=1;i<nums.length;i++) {  
            maxlast=Math.max(nums[i],maxlast+nums[i]);  
            maxfir=Math.max(maxfir,maxlast);  
        }  
        return maxfir;  
    }  
}
```

OUTPUT:



2. Write a program to print the multiplication table of number m up to n.

Sample Input:

M = 4

N = 5

Sample Output:

1x4=4

2x4=8

3x4=12

4x4=16

5x4=20

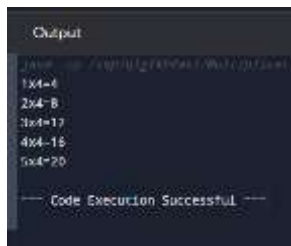
CODE:

```

public class MultiplicationTable {
    public static void main(String[] args) {
        int m = 4;
        int n = 5;
        for (int i = 1; i <= n; i++) {
            System.out.println(i + "x" + m + "=" + (i * m));
        }
    }
}

```

OUTPUT:



3. Write Java programs to implement multiple threads and apply join method for thread and thread has to be started after 500ms using sleep ().

CODE:

```

public class Join extends Thread
{
    public void run()
    {
        for(int i=1; i<=4; i++)
        {
            try
            {
                Thread.sleep(500);
            } catch (Exception e) { System.out.println(e); }
            System.out.println(i);
        }
    }
    public static void main(String args[])
    {
        Join t1 = new Join();
        Join t2 = new Join();
        Join t3 = new Join();
        t1.start();
        try
        {
            t1.join();
        }
    }
}

```

```

catch(Exception e)
{
System.out.println(e);
}
    t2.start();
    t3.start();
}
}

```

OUTPUT:



4. Generate a Java code that implements java selection and iteration statements. Use do while loop to process a menu selection. When a menu is selected, it should display the syntax of the selected statements.

CODE:

```

import java.util.Scanner;
public class MenuSelection {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int choice;
        do {
            System.out.println("Select an option:");
            System.out.println("1. If Statement");
            System.out.println("2. Switch Statement");
            System.out.println("3. For Loop");
            System.out.println("4. While Loop");
            System.out.println("0. Exit");
            System.out.print("Enter your choice: ");
            choice = scanner.nextInt();
            switch (choice) {
                case 1:
                    System.out.println("Syntax of If Statement: if (condition) { // code block }");
                    break;
                case 2:

```

```

        System.out.println("Syntax of Switch Statement: switch (expression) { case
value: // code block break; }");
        break;
    case 3:
        System.out.println("Syntax of For Loop: for (initialization; condition; update) {
// code block }");
        break;
    case 4:
        System.out.println("Syntax of While Loop: while (condition) { // code block }");
        break;
    case 0:
        System.out.println("Exiting...");
        break;
    default:
        System.out.println("Invalid choice. Please try again.");
    }
} while (choice != 0);

scanner.close();
}
}

```

OUTPUT:



```

Output
Select an option:
1. IF Statement
2. Switch Statement
3. For loop
4. While loop
5. Exit
Enter your choice: 1
Syntax of IF Statement: if (condition) { // code block }
Select an option:
1. IF Statement
2. Switch Statement
3. For loop
4. While loop
5. Exit
Enter your choice:

```

5. Create a simple generics class with type parameters for sorting values of different types.

CODE:

```

import java.util.Arrays;
public class GenericSort<T extends Comparable<T>> {
    private T[] array;
    public GenericSort(T[] array) {
        this.array = array;
    }
}

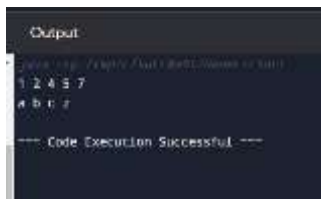
```

```

public void sort() {
    Arrays.sort(array);
}
public void printArray() {
    for (T element : array) {
        System.out.print(element + " ");
    }
    System.out.println();
}
}
public static void main(String[] args) {
    Integer[] intArray = {4, 2, 7, 1, 5};
    GenericSort<Integer> intSort = new GenericSort<>(intArray);
    intSort.sort();
    intSort.printArray();
    String[] strArray = {"z", "a", "c", "b"};
    GenericSort<String> strSort = new GenericSort<>(strArray);
    strSort.sort();
    strSort.printArray();
}
}

```

OUTPUT:



6. Create a class name 'overload'. Write a program to assign the values for two values by different number of arguments using a single function.

CODE:

```

public class Overload {
    void assignValues(int a) {
        System.out.println("Value 1: " + a);
    }
    void assignValues(int a, int b) {
        System.out.println("Value 1: " + a);
        System.out.println("Value 2: " + b);
    }
    public static void main(String[] args) {
        Overload obj = new Overload();
        obj.assignValues(10);
        obj.assignValues(20, 30);
    }
}

```

OUTPUT:



```
Output
=====
value 1: 10
value 1: 20
value 2: 30
=== Code Execution Successful ===
```

7. Write a Java Program to count the number of words in a string using Hash Map.

CODE:

```
import java.util.HashMap;
public class WordCounter {
    public static void main(String[] args) {
        String text = "Hello, how are you? This is a sample text with some words.";
        HashMap<String, Integer> wordCount = new HashMap<>();
        String[] words = text.split("\\s+");
        for (String word : words) {
            if (wordCount.containsKey(word)) {
                wordCount.put(word, wordCount.get(word) + 1);
            } else {
                wordCount.put(word, 1);
            }
        }
        System.out.println("Word Counts:");
        for (String word : wordCount.keySet()) {
            System.out.println(word + ": " + wordCount.get(word));
        }
    }
}
```

OUTPUT:



```
Output
=====
Word Counts:
how: 1
are: 1
you?: 1
This: 1
is: 1
text: 1
sample: 1
Hello: 1
with: 1
a: 1
some: 1
=== Code Execution Successful ===
```

8. Write a Java Program to read an email and password from excel sheet by retrieving the cell using `getRow()` and `getCell()` method

CODE:

9. Write a Java program to sort the given value using Hash Map.

CODE:

```
import java.util.*;

public class HashMapSort {
    public static void main(String[] args) {
        HashMap<Integer, String> map = new HashMap<>();
        map.put(3, "Apple");
        map.put(1, "Banana");
        map.put(2, "Orange");

        System.out.println("Before Sorting:");
        for (Map.Entry<Integer, String> entry : map.entrySet()) {
            System.out.println(entry.getKey() + ": " + entry.getValue());
        }

        Map<Integer, String> sortedMap = new TreeMap<>(map);

        System.out.println("\nAfter Sorting:");
        for (Map.Entry<Integer, String> entry : sortedMap.entrySet()) {
            System.out.println(entry.getKey() + ": " + entry.getValue());
        }
    }
}
```

OUTPUT:



```
Output
Before Sorting:
1: Banana
2: Orange
3: Apple

After Sorting:
1: Banana
2: Orange
3: Apple

--- Code Execution Successful ---
```

10. Write a Java program to find distinct characters and their count in a string.

CODE:

```
import java.util.HashMap;

public class DistinctCharactersCount {
    public static void main(String[] args) {
```

```

String str = "hello world";
HashMap<Character, Integer> charCountMap = new HashMap<>();
for (char c : str.toCharArray()) {
    if (charCountMap.containsKey(c)) {
        charCountMap.put(c, charCountMap.get(c) + 1);
    } else {
        charCountMap.put(c, 1);
    }
}
System.out.println("Distinct characters in the string and their counts:");
for (char c : charCountMap.keySet()) {
    System.out.println(c + ": " + charCountMap.get(c));
}
}
}

```

OUTPUT:



11. Write a program to print all the unique characters in a String. For instance, if the input string is “abcb”, the output will be the characters ‘a’ and ‘c’ as they are unique. The character ‘b’ repeats twice and so it will not be printed.

CODE:

```

import java.util.HashSet;
import java.util.Set;
public class UniqueCharacters {
    public static void main(String[] args) {
        String input = "abcb";
        Set<Character> uniqueChars = new HashSet<>();
        for (char c : input.toCharArray()) {
            if (input.indexOf(c) == input.lastIndexOf(c)) {
                uniqueChars.add(c);
            }
        }

        for (char uniqueChar : uniqueChars) {

```



```
        System.out.print(uniqueChar + " ");  
    }  
}  
}
```

OUTPUT:



```
Output  
root@kali:~/Desktop/Java/UniqueCharacters  
h e  
--- Code Execution Successful ---
```