# Java Programming Assignment

Name: Dinesh Kumar K

Registration Number: RA2411026010127

1. 1. Write a Java method to find the second largest number in an array.

public static int findSecondLargest(int[] arr) {  
 int first = Integer.MIN\_VALUE;  
 int second = Integer.MIN\_VALUE;  
 for (int num : arr) {  
 if (num > first) {  
 second = first;  
 first = num;  
 } else if (num > second && num != first) {  
 second = num;  
 }  
 }  
 return second;  
}

1. 2. Create a Java method for a library management system that checks if a requested book is available in the library's catalog.

public static String isBookAvailable(String[] catalog, String bookName) {  
 for (String book : catalog) {  
 if (book.equalsIgnoreCase(bookName)) {  
 return "Book is available";  
 }  
 }  
 return "Book is not available";  
}

1. 3. Write a method that takes an array and returns true if there is a 3 next to a 3 somewhere.

public static boolean has33(int[] nums) {  
 for (int i = 0; i < nums.length - 1; i++) {  
 if (nums[i] == 3 && nums[i + 1] == 3) {  
 return true;  
 }  
 }  
 return false;  
}

1. 4. Write a Java program using a method to print a pyramid pattern of stars for a given number of rows.

public static void printPyramid(int rows) {  
 for (int i = 1; i <= rows; i++) {  
 for (int j = i; j < rows; j++) {  
 System.out.print(" ");  
 }  
 for (int k = 1; k <= (2 \* i - 1); k++) {  
 System.out.print("\*");  
 }  
 System.out.println();  
 }  
}

1. 5. Write a method to find the frequency of each element in an array.

public static void findFrequency(int[] arr) {  
 Map<Integer, Integer> freqMap = new HashMap<>();  
 for (int num : arr) {  
 freqMap.put(num, freqMap.getOrDefault(num, 0) + 1);  
 }  
 for (Map.Entry<Integer, Integer> entry : freqMap.entrySet()) {  
 System.out.println(entry.getKey() + "->" + entry.getValue());  
 }  
}

1. 6. Write a Java method to reverse each word in a given sentence.

public static String reverseWords(String sentence) {  
 String[] words = sentence.split(" ");  
 StringBuilder result = new StringBuilder();  
 for (String word : words) {  
 result.append(new StringBuilder(word).reverse().toString()).append(" ");  
 }  
 return result.toString().trim();  
}

1. 7. Write a program using a method to print a specific number pattern.

public static void printPattern(int n) {  
 for (int i = 1; i <= n; i++) {  
 for (int j = 1; j <= i; j++) {  
 System.out.print(j + " ");  
 }  
 System.out.println();  
 }  
}

1. 8. Write a Java method to find and return the common elements between two arrays.

public static List<Integer> findCommonElements(int[] arr1, int[] arr2) {  
 List<Integer> result = new ArrayList<>();  
 Set<Integer> set = new HashSet<>();  
 for (int num : arr1) {  
 set.add(num);  
 }  
 for (int num : arr2) {  
 if (set.contains(num)) {  
 result.add(num);  
 }  
 }  
 return result;  
}

1. 9. Write a Java program with a method to sort an array using the bubble sort algorithm.

public static void bubbleSort(int[] arr) {  
 int n = arr.length;  
 for (int i = 0; i < n - 1; i++) {  
 for (int j = 0; j < n - i - 1; j++) {  
 if (arr[j] > arr[j + 1]) {  
 int temp = arr[j];  
 arr[j] = arr[j + 1];  
 arr[j + 1] = temp;  
 }  
 }  
 }  
}

1. 10. Write a Java method to print a right-aligned triangle of numbers for n rows.

public static void printRightAlignedTriangle(int n) {  
 for (int i = 1; i <= n; i++) {  
 for (int j = i; j < n; j++) {  
 System.out.print(" ");  
 }  
 for (int j = 1; j <= i; j++) {  
 System.out.print(j);  
 }  
 System.out.println();  
 }  
}