Sachin Singh
M24CSE033

Q. Same as above question, a variable "d" is also given in the question denoting the divisor. Find the maximum sum of a subsequence such that no two numbers in the subsequence are adjacent in the array, and the sum of the selected subsequence should be divisible by a given number d.

So, for example, d = 10 then output will be 110 for a given array in Q2.

Explanation:

1. **findKthLargest**: This function will return the k-th largest element in an array after sorting it in descending order.

2. **maxNonAdjacentSumDivisibleByD**: The function returns the maximum sum of a subsequence in which no two elements are adjacent, and the sum is divisible by d. Use two variables: include (to include the current element in the sum) and exclude (to exclude the current element). If it is not divisible by d, then it will return the sum after adding the k-th largest element.

3. **Main Function**: We define an array arr[], find the k-th largest element, then call the function maxNonAdjacentSumDivisibleByD to calculate the maximum sum divisible by d.

Example

```
int arr[] = {5, 5, 10, 100, 10, 5};  // Example array
int n = sizeof(arr) / sizeof(arr[0]);
int k = 2;  // k-th largest element
int d = 10; // Given divisor

// Step 1: Find the k-th largest element
int kthLargest = findKthLargest(arr, n, k);
printf("The %d-th largest element is: %d\n", k, kthLargest);

// Step 2: Find the maximum sum of subsequence including the k-th largest element
// and divisible by d
int maxSum = maxNonAdjacentSumDivisibleByD(arr, n, d, kthLargest);
printf("Maximum sum of subsequence divisible by %d: %d\n", d, maxSum);
```

Output

```
********** Sachin Singh **********
********** M24CSE033 **********
--------------------------------

The 2-th largest element is: 10
Maximum sum of subsequence divisible by 10: 110


...Program finished with exit code 0
Press ENTER to exit console.
```