Q1 –

```cpp
#include <iostream>
#include <cmath>

using namespace std;

int main()
{
    int n;
    double sum = 0.0;
    cout << "Enter the number of terms (n): ";
    cin >> n;

    for (int i = 1; i <= n; i++)
    {
        double term = 1.0 / pow(i, i);
        if (i % 2 == 0)
            term *= -1;
        sum += term;
    }

    cout << "Sum of first " << n << " terms: " << sum << endl;

    return 0;
}
```

Output –

```
Enter the number of terms (n): 5
Sum of first 5 terms: 0.783451
```

Q2 –

```cpp
#include <iostream>

int main()
{
    int arr[] = {1, 2, 2, 3, 4, 4, 5, 6, 6, 7};
    int n = sizeof(arr) / sizeof(arr[0]);

    std::cout << "Original array: ";
    for (int i = 0; i < n; i++)
    {
        std::cout << arr[i] << " ";
```

```
    }
    std::cout << std::endl;

    int newSize = 0;
    for (int i = 0; i < n; i++)
    {
        bool isDuplicate = false;
        for (int j = 0; j < newSize; j++)
        {
            if (arr[i] == arr[j])
            {
                isDuplicate = true;
                break;
            }
        }
        if (!isDuplicate)
        {
            arr[newSize] = arr[i];
            newSize++;
        }
    }

    std::cout << "Array after removing duplicates: ";
    for (int i = 0; i < newSize; i++)
    {
        std::cout << arr[i] << " ";
    }
    std::cout << std::endl;

    return 0;
}
```

Output –

```
Original array: 1 2 2 3 4 4 5 6 6 7
Array after removing duplicates: 1 2 3 4 5 6 7
```

Q3 –

```
#include <iostream>

int main()
{
    int arr1[] = {1, 3, 5, 7};
    int arr2[] = {2, 4, 6, 8};
```

```
    int n1 = sizeof(arr1) / sizeof(arr1[0]);
    int n2 = sizeof(arr2) / sizeof(arr2[0]);
    int merged[n1 + n2];

    std::cout << "First array: ";
    for (int i = 0; i < n1; i++)
    {
        std::cout << arr1[i] << " ";
    }
    std::cout << std::endl;

    std::cout << "Second array: ";
    for (int i = 0; i < n2; i++)
    {
        std::cout << arr2[i] << " ";
    }
    std::cout << std::endl;

    int i = 0, j = 0, k = 0;
    while (i < n1 && j < n2)
    {
        if (arr1[i] < arr2[j])
        {
            merged[k++] = arr1[i++];
        }
        else
        {
            merged[k++] = arr2[j++];
        }
    }

    while (i < n1)
    {
        merged[k++] = arr1[i++];
    }

    while (j < n2)
    {
        merged[k++] = arr2[j++];
    }

    std::cout << "Merged array: ";
    for (int i = 0; i < n1 + n2; i++)
    {
        std::cout << merged[i] << " ";
```

```
    }
    std::cout << std::endl;

    return 0;
}
```

Output –

```
First array: 1 3 5 7
Second array: 2 4 6 8
Merged array: 1 2 3 4 5 6 7 8
```