NAME:SAYALI JIVAN CHAUDHARI

ROLL NO.:14

PRN NO.2023015400005055

1)implementation of programs based on polynomials addition and subtraction

//adding two polynomials using arrays

```cpp
// Simple C++ program to add two polynomials
#include <iostream>
using namespace std;

// A utility function to return maximum of two integers
int max(int m, int n) { return (m > n) ? m : n; }

// A[] represents coefficients of first polynomial
// B[] represents coefficients of second polynomial
// m and n are sizes of A[] and B[] respectively
```

```cpp
int* add(int A[], int B[], int m, int n)
{
    int size = max(m, n);
    int* sum = new int[size];

    // Initialize the product polynomial
    for (int i = 0; i < m; i++)
        sum[i] = A[i];

    // Take every term of first polynomial
    for (int i = 0; i < n; i++)
        sum[i] += B[i];

    return sum;
}

// A utility function to print a polynomial
void printPoly(int poly[], int n)
{
```

```cpp
    for (int i = 0; i < n; i++) {

        cout << poly[i];

        if (i != 0)

            cout << "x^" << i;

        if (i != n - 1)

            cout << " + ";

    }

}

// Driver program to test above functions
int main()
{
    // The following array represents polynomial 5 +
10x^2 +
    // 6x^3
    int A[] = { 5, 0, 10, 6 };

    // The following array represents polynomial 1 + 2x +
    // 4x^2
```

```cpp
    int B[] = { 1, 2, 4 };
    int m = sizeof(A) / sizeof(A[0]);
    int n = sizeof(B) / sizeof(B[0]);

    cout << "First polynomial is \n";
    printPoly(A, m);
    cout << "\nSecond polynomial is \n";
    printPoly(B, n);

    int* sum = add(A, B, m, n);
    int size = max(m, n);

    cout << "\nsum polynomial is \n";
    printPoly(sum, size);

    return 0;
}
```

//2)subtracting two polynomials

// Simple C++ program to add two polynomials

```cpp
#include <iostream>
using namespace std;

// A utility function to return maximum of two integers
int max(int m, int n) { return (m > n) ? m : n; }


// A[] represents coefficients of first polynomial
// B[] represents coefficients of second polynomial
// m and n are sizes of A[] and B[] respectively
int* add(int A[], int B[], int m, int n)
{
    int size = max(m, n);
    int* sum = new int[size];

    // Initialize the product polynomial
    for (int i = 0; i < m; i++)
        sum[i] = A[i];

    // Take every term of first polynomial
```

```cpp
    for (int i = 0; i < n; i++)
        sum[i] -= B[i];

    return sum;
}


// A utility function to print a polynomial
void printPoly(int poly[], int n)
{
    for (int i = 0; i < n; i++) {
        cout << poly[i];
        if (i != 0)
            cout << "x^" << i;
        if (i != n - 1)
            cout << " + ";
    }
}


// Driver program to test above functions
```

```cpp
int main()
{
    // The following array represents polynomial 5 + 10x^2 +
    // 6x^3
    int A[] = { 5, 0, 10, 6 };

    // The following array represents polynomial 1 + 2x +
    // 4x^2
    int B[] = { 1, 2, 4 };
    int m = sizeof(A) / sizeof(A[0]);
    int n = sizeof(B) / sizeof(B[0]);

    cout << "First polynomial is \n";
    printPoly(A, m);
    cout << "\nSecond polynomial is \n";
    printPoly(B, n);

    int* sum = add(A, B, m, n);
```

```cpp
    int size = max(m, n);

    cout << "\nsub polynomial is \n";
    printPoly(sum, size);

    return 0;
}
```