PolyLLMai… AddPolyL… Submit

Explorer Debugger Plots

```
struct polynomial
{
    int coeff;
    int exp;
    struct polynomial *next;
};

typedef struct polynomial *poly;


poly addTerm(poly head, poly temp)
{
    poly p1 ;
    p1 = head;
    poly p2;
    p2 = head;

    if( p1 == NULL )
    {
        head = temp;
    }
    else
    {
        while( p1 != NULL && p1 -> exp > temp -> exp )
        {
            p2 = p1;
            p1 = p1->next;
        }

        if( p1 == NULL )
        {
            p2->next = temp;
        }

        else if( p1->exp == temp->exp )
        {
            p1->coeff = p1->coeff + temp->coeff;
        }

        else if( p1->exp < temp->exp )
        {
            if( p1 == p2 )
            {
```

‹ Prev Reset Submit Next ›

```
                temp->next = p1;
                head = temp;
            }
            else
            {
                temp->next = p1;
                p2->next = temp;
            }

        }

    }


    return head;

}


void print(poly head) {
    poly temp = head;
    while(temp != NULL) {
        printf("%d X^ %d ---> ", temp -> coeff, temp -> exp);
        temp = temp -> next;
    }
    printf("NULL\n");
}



poly add(poly head1, poly head2)
{
    poly ans = NULL;

    poly temp1 = head1;

    while( temp1!=NULL )
    {
        poly temp = malloc( sizeof( struct polynomial ) );
        temp->next = NULL;
        temp->coeff = temp1->coeff;
        temp->exp = temp1->exp;

        ans = addTerm(ans , temp);

```

```
91          temp1 = temp1->next;
92      }
93
94      poly temp2 = head2;
95
96      while( temp2 != NULL )
97      {
98          poly temp = malloc( sizeof(struct polynomial) );
99          temp->next = NULL;
100         temp->coeff = temp2->coeff;
101         temp->exp = temp2->exp;
102
103         ans = addTerm( ans,temp );
104
105         temp2 = temp2->next;
106     }
107
108     return ans;
109
110 }
111
112
```

Terminal Test cases