Assignment 1

Name = Om D Tambat

Enrollment No. = BT22CSHO42

```
#include <stdio.h>
#include <stdlib.h>

struct node
{
   int coe;
   int exp;
   struct node *link;
};

struct list
{
   struct node *head;
   struct node *temp;
};

struct list *createlist(struct list *list1)
{
   list1->head = NULL;
   list1->temp = NULL;
   return list1;
}
```

```
void newnode(int coe, int exp, struct list *list1)
{
   struct node *newNode = (struct node *)malloc(sizeof(struct node));
   newNode->coe = coe;
   newNode->exp = exp;
   newNode->link = NULL;

   if (list1->head == NULL)
   {
      list1->head = newNode;
      list1->temp = newNode;
   }
   else
   {
      list1->temp->link = newNode;
      list1->temp = newNode;
   }
}

void display(struct list *list1)
{
   struct node *temp1 = list1->head;
   if (list1->head == NULL)
   {
      printf("list is empty");
```

```
        return;
    }

    printf("polynomial is\n");
    while (temp1 != NULL)
    {
        if (temp1->coe < 0)
        {
            printf("%dx^%d ", temp1->coe, temp1->exp);
        }
        else
        {
            printf("+%dx^%d ", temp1->coe, temp1->exp);
        }
        temp1 = temp1->link;
    }
    printf("\n");
}

struct list *add(struct list *list1, struct list *list2)
{
    struct list *result = (struct list *)malloc(sizeof(struct list));
    createlist(result);

    struct node *temp = NULL;
    struct node *list1Node = list1->head;
```

```
struct node *list2Node = list2->head;

while (list1Node != NULL && list2Node != NULL)
{
   int newCoe;
   int newExp;

   if (list1Node->exp == list2Node->exp)
   {
      newCoe = list1Node->coe + list2Node->coe;
      newExp = list1Node->exp;
      list1Node = list1Node->link;
      list2Node = list2Node->link;
   }
   else if (list1Node->exp > list2Node->exp)
   {
      newCoe = list1Node->coe;
      newExp = list1Node->exp;
      list1Node = list1Node->link;
   }
   else
   {
      newCoe = list2Node->coe;
      newExp = list2Node->exp;
      list2Node = list2Node->link;
   }
```

```
        newnode(newCoe, newExp, result);
    }

    while (list1Node != NULL)
    {
        newnode(list1Node->coe, list1Node->exp, result);
        list1Node = list1Node->link;
    }

    while (list2Node != NULL)
    {
        newnode(list2Node->coe, list2Node->exp, result);
        list2Node = list2Node->link;
    }

    return result;
}

struct list *multiply(struct list *list1, struct list *list2)
{
    struct list *result = (struct list *)malloc(sizeof(struct list));
    createlist(result);

    struct node *list1Node = list1->head;
```

```
while (list1Node != NULL)
{
   struct node *list2Node = list2->head;

   while (list2Node != NULL)
   {
      int newCoe = list1Node->coe * list2Node->coe;
      int newExp = list1Node->exp + list2Node->exp;

      struct node *temp = result->head;
      struct node *prev = NULL;
      while (temp != NULL && temp->exp > newExp)
      {
         prev = temp;
         temp = temp->link;
      }

      if (temp == NULL)
      {
         newnode(newCoe, newExp, result);
      }
      else if (temp->exp < newExp)
      {
         struct node *newNode = (struct node *)malloc(sizeof(struct node));
         newNode->coe = newCoe;
         newNode->exp = newExp;
```

```
            newNode->link = temp;

            if (prev == NULL)
            {
               result->head = newNode;
            }
            else
            {
               prev->link = newNode;
            }
         }
         else if (temp->exp == newExp)
         {
            temp->coe += newCoe;
         }

         list2Node = list2Node->link;
      }

      list1Node = list1Node->link;
   }

   return result;
}

int main()
```

```
{
    int coe, n;
    struct list list1, list2, *list3, *list4;

    createlist(&list1);
    createlist(&list2);

    printf("Enter the degree of the first polynomial: ");
    scanf("%d", &n);

    for (int i = n; i >= 0; i--)
    {
        printf("Enter coefficient of x^%d: ", i);
        scanf("%d", &coe);
        newnode(coe, i, &list1);
    }

    printf("Enter the degree of the second polynomial: ");
    scanf("%d", &n);
    for (int i = n; i >= 0; i--)
    {
        printf("Enter coefficient of x^%d: ", i);
        scanf("%d", &coe);
        newnode(coe, i, &list2);
    }
```

```
    display(&list1);
    display(&list2);

    list3 = add(&list1, &list2);
    printf("Sum: ");
    display(list3);

    list4 = multiply(&list1, &list2);
    printf("Product: ");
    display(list4);

    return 0;
}
```

Output →

Output Clear

```
/tmp/8kwBtTq2kb.o
Enter the degree of the first polynomial: 1
Enter coefficient of x^1: 2
Enter coefficient of x^0: 1
Enter the degree of the second polynomial: 2
Enter coefficient of x^2: 1
Enter coefficient of x^1: 2
Enter coefficient of x^0: 1
polynomial is
+2x^1 +1x^0
polynomial is
+1x^2 +2x^1 +1x^0
Sum: polynomial is
+1x^2 +4x^1 +2x^0
Product: polynomial is
+2x^3 +5x^2 +4x^1 +1x^0
```