DS LAB Week 2

Name: Adithya M SRN: PES1UG20CS621 SECTION: K

1) Add Polynomials

```c
#include<stdio.h>
#include<stdlib.h>
typedef struct node
{
    int coeff;
    int powx;
    int powy;
    struct node *next;
}node;
typedef struct llist
{
    node *head;
}llist;
void swap_node(llist *,int, int);
void disp(llist *);
void add_pol(llist *res,llist *p1,llist *p2);
int main()

    {
        llist list_of_polynomials[2];
        for(int i=0;i<2;i++)
        {
                printf("Enter the number of terms you want to enter in the polynomial %d:\n",i+1);
                int n;
                scanf("%d",&n);
                list_of_polynomials[i].head = malloc(sizeof(node));
                if(n==0)
                {
                    list_of_polynomials[i].head = NULL;
                }
                if(n>0)
                {
                    printf("Enter the co-efficient and the power of x and y respectively in the first polynomial\n");
                    scanf("%d %d %d",&list_of_polynomials[i].head->coeff,&list_of_polynomials[i].head->powx,&list_of_polynomials[i].head->powy);
                    list_of_polynomials[i].head->next = (node *)malloc(sizeof(node));
                    node *q = list_of_polynomials[i].head->next;node *p;
                    for(int i=0;i<n-1;i++)
                    {
                        scanf("%d %d %d",&q->coeff,&q->powx,&q->powy);
                        q->next = malloc(sizeof(node));
                        p = q;
                        q = q->next;

                    }
                    p->next = NULL;
                    free(q);
                }
        }
        printf("the polynomials are:\n");
        for(int i=0;i<2;i++)
        {
```

```
        for(int i=0;i<2;i++)
        {
            printf("polynomial %d\n",i+1);
            disp(&list_of_polynomials[i]);
        }
        llist res;
        res.head = malloc(sizeof(node));
        add_pol(&res,&list_of_polynomials[0],&list_of_polynomials[1]);
        printf("the resultant of the addition of the given two polynomials is:\n");
        disp(&res);
    }

void add_pol(llist *res,llist *p1,llist *p2)
{
    node *temp1 = p1->head,*temp2 = p2->head,*temp_res = res->head,*var;int k = 0;
    while(temp1!=NULL)
    {
        while(temp2!=NULL)
        {
            if((temp1->powx == temp2->powx) && (temp1->powy == temp2->powy))
            {
                temp_res->coeff = (temp1->coeff)+(temp2->coeff);
                temp_res->powx = temp1->powx;
                temp_res->powy = temp1->powy;
                var = temp_res;
                temp_res->next = malloc(sizeof(node));
                temp_res = temp_res->next;
                k = 1;
            }
            temp2 = temp2->next;
        }
        if(k == 0)
        {
            temp_res->coeff = temp1->coeff;
            temp_res->powx = temp1->powx;
            temp_res->powy = temp1->powy;
            var = temp_res;
            temp_res->next = malloc(sizeof(node));
            temp_res = temp_res->next;
            k = 0;
        }
        temp2 = p2->head;
        temp1 = temp1->next;
    }
    var->next = NULL;
}
```

```
void disp(llist *a)
{
    node *p=a->head;
    while(p!=NULL)
    {
        printf("%dx^%dy^%d\n",p->coeff,p->powx,p->powy);
        p = p->next;
    }
}
```

Output:

```
PS C:\Users\adith\Documents\C Programs\week 2> cd "c:
Enter the number of terms you want to enter in the po
2
Enter the co-efficient and the power of x and y respe
2 3 5
6 4 2
Enter the number of terms you want to enter in the po
2
Enter the co-efficient and the power of x and y respe
4 5 2
3 5 6
the polynomials are:
polynomial 1
2x^3y^5
6x^4y^2
polynomial 2
4x^5y^2
3x^5y^6
the resultant of the addition of the given two polyno
2x^3y^5
6x^4y^2
PS C:\Users\adith\Documents\C Programs\week 2>
```

2) Swap sll

```c
#include<stdio.h>
#include"swapsllh.h"
int main()
{
        struct node* start = NULL;
        addnode(&start , 7);
        addnode(&start , 6);
        addnode(&start , 5);
        addnode(&start , 4);
        addnode(&start , 3);
        addnode(&start , 2);
        addnode(&start , 1);

        printf("\n linked list before caliing swapnodes()");
        printlist(start);
        swapnodes(&start,4,3);
        printf("\n linked list after caliing swapnodes()");
        printlist(start);
        printf("/n");
        return 0;
}
```

```c
struct node{
        int data;
        struct node *next;
        };

void swapnodes(struct node** head_ref,int x,int y);
void addnode(struct node** head_ref,int new_data);
void printlist(struct node* node);
```

```c
#include<stdio.h>
#include<stdlib.h>
#include"swapsllh.h"
void swapnodes(struct node** head_ref,int x,int y)
{
        if(x==y)
                return;
        struct node *prevX = NULL ,*presX = *head_ref;
        while(presX && presX-> data != x)
        {
                prevX = presX;
                presX = presX -> next;
        }
        //Search for y(keeping track of prevY and presY
        struct node *prevY = NULL ,*presY = *head_ref;
        while(presY && presY-> data !=y)
        {
                prevY = presY;
                presY = presY -> next;
        }

        if(presX == NULL || presY == NULL)
        {
                return;

        if(presX != NULL)
                prevX -> next =presY;
        else
                *head_ref = presY;

        if(presY != NULL)
                prevY -> next =presX;
        else
                *head_ref = presX;


        //swap next pointers

         struct node* temp = presY -> next;
         presY -> next = presX -> next;
         presX -> next = temp;
         }
         }

         //Function to add a node at the beginning of the list
void addnode(struct node** head_ref,int new_data)
{
        struct node* new_node =  (struct node*)malloc(sizeof(struct node));
        new_node -> data = new_data;
        new_node -> next = (*head_ref);
        (*head_ref) = new_node;
```

Output:

```c
void printlist(struct node* node)
{
        while(node!= NULL)
        {
                printf("%d",node -> data);
                node = node -> next;
                }
                }

```