NAME- Aniket Kalbhor

PRN- 12210601

ROLL- CSB-48

CODE- **3. Write a program to implement Polynomial addition using linked list.**

#include <stdio.h>

#include <stdlib.h>

struct poly

{

int coeff;

int exp;

struct poly\* next;

};

struct poly\* add(struct poly\* p1, struct poly\* p2)

{

struct poly\* head3 = NULL;

struct poly\* p3 = NULL;

while (p1 != NULL && p2 != NULL)

{

if (head3 == NULL)

{

head3 = p3 = (struct poly\*)malloc(sizeof(struct poly));

}

else

{

p3->next = (struct poly\*)malloc(sizeof(struct poly));

p3 = p3->next;

p3->next = NULL;

}

if (p1->exp == p2->exp)

{

p3->coeff = p1->coeff + p2->coeff;

p3->exp = p1->exp;

p1 = p1->next;

p2 = p2->next;

}

else if (p1->exp > p2->exp)

{

p3->coeff = p1->coeff;

p3->exp = p1->exp;

p1 = p1->next;

}

else

{

p3->coeff = p2->coeff;

p3->exp = p2->exp;

p2 = p2->next;

}

}

while (p1 != NULL)

{

if (head3 == NULL)

{

head3 = p3 = (struct poly\*)malloc(sizeof(struct poly));

p3->next = NULL;

}

else

{

p3->next = (struct poly\*)malloc(sizeof(struct poly));

p3 = p3->next;

p3->next = NULL;

}

if (p1->exp = p2->exp)

{

p3->coeff = p1->coeff + p2->coeff;

p3->exp = p1->exp;

p1 = p1->next;

p2 = p2->next;

}

else if (p1->exp > p2->exp)

{

p3->coeff = p1->coeff;

p3->exp = p1->exp;

p1 = p1->next;

}

else

{

p3->coeff = p2->coeff;

p3->exp = p2->exp;

p2 = p2->next;

}

p3->coeff = p1->coeff;

p3->exp = p1->exp;

p1 = p1->next;

}

while (p2 != NULL)

{

if (head3 == NULL)

{

head3 = p3 = (struct poly\*)malloc(sizeof(struct poly));

p3->next = NULL;

}

else

{

p3->next = (struct poly\*)malloc(sizeof(struct poly));

p3 = p3->next;

p3->next = NULL;

}

if (p1->exp = p2->exp)

{

p3->coeff = p1->coeff + p2->coeff;

p3->exp = p1->exp;

p1 = p1->next;

p2 = p2->next;

}

else if (p1->exp > p2->exp)

{

p3->coeff = p1->coeff;

p3->exp = p1->exp;

p1 = p1->next;

}

else

{

p3->coeff = p2->coeff;

p3->exp = p2->exp;

p2 = p2->next;

}

p3->coeff = p2->coeff;

p3->exp = p2->exp;

p2 = p2->next;

}

return head3;

};

struct poly\* accept\_poly(int n)

{

struct poly\* head, \*p;

head = p = (struct poly\*)malloc(sizeof(struct poly));

p->next = NULL;

printf("Enter first coefficient and power\n");

scanf("%d %d",&p->coeff, &p->exp);

for (int i = 1;i < n;i++)

{

p->next = (struct poly\*)malloc(sizeof(struct poly));

p = p->next;

p->next = NULL;

scanf("%d %d",&p->coeff, &p->exp);

}

return head;

};

void printPoly(struct poly\* p)

{

while (p != NULL)

{

printf("%dx^%d + ",p->coeff,p->exp);

p = p->next;

}

printf("0");

printf("\n");

}

int main()

{

struct poly\* poly1;

struct poly\* poly2;

struct poly\* poly3;

int n1, n2;

printf("Enter number of terms in each polynomial\n");

scanf("%d %d",&n1, &n2);

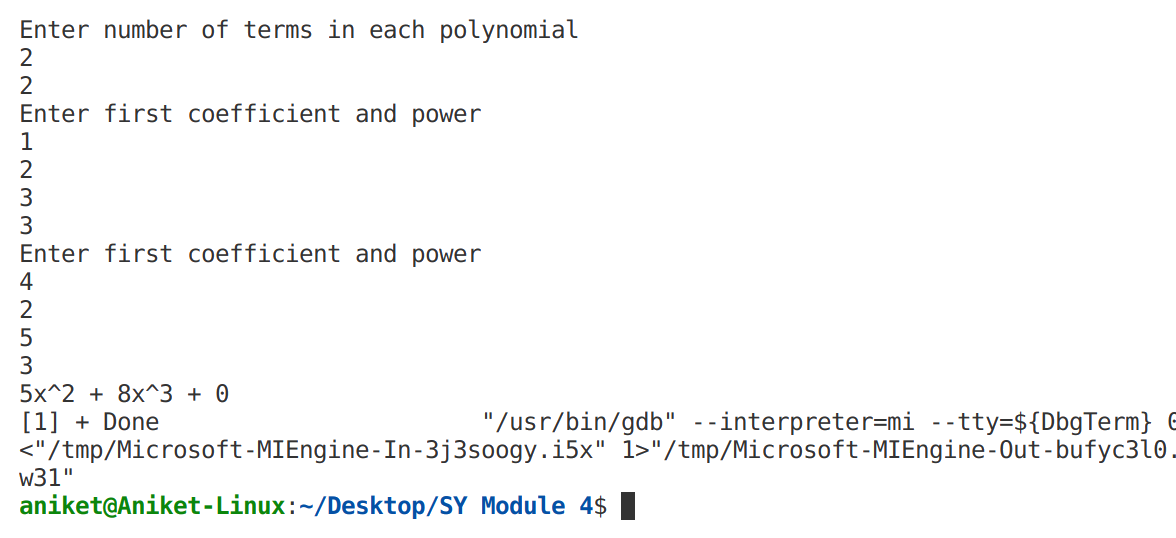
poly1 = accept\_poly(n1);

poly2 = accept\_poly(n2);

poly3 = add(poly1, poly2);

printPoly(poly3);

}

OUTPUT-