

VARUN BHARGAVA – 241010282

DATA STRUCTURES TASK-7

Task 01: Queue:

(<https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab7/lab7-1.c>)

Use a queue to convert a given positive integer into its binary representation.

Steps:

1. Input a decimal number (e.g., 19).
2. Repeatedly divide the number by 2, store remainders in a queue.
3. Dequeue elements to print the binary equivalent.

The image shows two screenshots of a code editor interface, likely Visual Studio Code, displaying different implementations of a queue-based conversion algorithm. Both screenshots show the code in the left pane and a terminal window in the right pane.

Screenshot 1 (Top):

```
lab7 > C lab7-1.c > main()
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 #define max 100
5
6 struct Queue
7 {
8     int data[max];
9     int front, rear;
10};
11
12 void enqueue(struct Queue *x, int a)
13 {
14     if (x->rear < max - 1)
15         x->data[++(x->rear)] = a;
16 }
17
18 int dequeue(struct Queue *x)
19 {
20     if (x->front <= x->rear)
21         return x->data[(x->front)++];
22     return -1;
23 }
24
25 int isEmpty(struct Queue *x)
26 {
27     return (x->front > x->rear);
28 }
29
30 int main()
31 {
32     int n;
33     printf("Enter a number: ");
34     scanf("%d", &n);
35 }
```

Terminal output:

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa> cd "c:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7\" ; if ($?) { gcc lab7-1.c -o lab7-1 } ; if ($?) { .\lab7-1 }
Enter a number: 19
The binary representation of 19 is: 10011
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7>
```

Screenshot 2 (Bottom):

```
lab7 > C lab7-1.c > main()
25 int isEmpty(struct Queue *x)
26
27 int main()
28 {
29     int n;
30     printf("Enter a number: ");
31     scanf("%d", &n);
32
33     struct Queue x;
34     x.front = 0;
35     x.rear = -1;
36
37     int temp = n;
38     if (temp == 0)
39     {
40         enqueue(&x, 0);
41     }
42     else
43     {
44         while (temp > 0)
45         {
46             enqueue(&x, temp % 2);
47             temp /= 2;
48         }
49
50         printf("The binary representation of %d is: ", n);
51         for (int i = x.rear; i >= x.front; i--)
52         {
53             printf("%d", x.data[i]);
54         }
55         printf("\n");
56     }
57     return 0;
58 }
```

Terminal output:

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa> cd "c:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7\" ; if ($?) { gcc lab7-1.c -o lab7-1 } ; if ($?) { .\lab7-1 }
Enter a number: 19
The binary representation of 19 is: 10011
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7>
```

Task 02: Doubly Linked List:

(<https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab7/lab7-2.c>)

1. Use a doubly linked list to add two large integers (each digit as one node).
2. Steps:
 1. Store each digit of the two numbers in separate doubly linked lists.
 2. Perform addition digit by digit from the rightmost (using backward traversal).
 3. Handle carry properly and display the result list.

The screenshot shows a code editor with two tabs: lab7-1.c U and lab7-2.c X. The lab7-2.c tab contains the following C code:

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4
5 struct Node
6 {
7     int digit;
8     struct Node *prev, *next;
9 };
10
11 void append(struct Node **head, int digit)
12 {
13     struct Node *newNode = (struct Node *)malloc(sizeof(struct Node));
14     newNode->digit = digit;
15     newNode->next = NULL;
16     newNode->prev = NULL;
17     if (*head == NULL)
18     {
19         *head = newNode;
20         return;
21     }
22     struct Node *temp = *head;
23     while (temp->next)
24     {
25         temp = temp->next;
26     }
27     temp->next = newNode;
28     newNode->prev = temp;
29 }
30
31 void printlist(struct Node *head)
32 {
33     struct Node *temp = head;
34     while (temp)
35     {
36         printf("%d", temp->digit);
37         temp = temp->next;
38     }
39     printf("\n");
40 }
41
42 struct Node *addLists(struct Node *head1, struct Node *head2)
43 {
44     struct Node *temp1 = head1;
45     struct Node *temp2 = head2;
46     int carry = 0;
47     struct Node *head3 = NULL;
48     struct Node *temp3 = NULL;
49     while (temp1 != NULL && temp2 != NULL)
50     {
51         int sum = temp1->digit + temp2->digit + carry;
52         carry = sum / 10;
53         sum = sum % 10;
54         if (head3 == NULL)
55         {
56             head3 = temp3 = (struct Node *)malloc(sizeof(struct Node));
57             temp3->digit = sum;
58             temp3->next = NULL;
59             temp3->prev = NULL;
60         }
61         else
62         {
63             temp3->next = (struct Node *)malloc(sizeof(struct Node));
64             temp3->next->prev = temp3;
65             temp3->next->digit = sum;
66             temp3->next->next = NULL;
67             temp3 = temp3->next;
68         }
69         temp1 = temp1->next;
70         temp2 = temp2->next;
71     }
72     if (temp1 == NULL && temp2 == NULL)
73     {
74         if (carry == 1)
75         {
76             temp3->next = (struct Node *)malloc(sizeof(struct Node));
77             temp3->next->prev = temp3;
78             temp3->next->digit = carry;
79             temp3->next->next = NULL;
80             temp3 = temp3->next;
81         }
82     }
83     else if (temp1 == NULL)
84     {
85         temp3->next = temp1;
86         temp1->prev = temp3;
87     }
88     else if (temp2 == NULL)
89     {
90         temp3->next = temp2;
91         temp2->prev = temp3;
92     }
93     return head3;
94 }
```

The terminal window shows the execution of the program. It first compiles the code with `gcc lab7-2.c -o lab7-1`. Then it enters a loop where it prompts for two large integers. The first integer is 19, which is printed in binary as 10011. The second integer is 123413243421. The sum of these two integers is calculated as 3374657776674, which is then printed.

```

lab7 > C:\lab7-1.c U C:\lab7-2.c U
lab7 > C:\lab7-2.c > @ append(Node **,int)
40 struct Node *addlists(struct Node *head1, struct Node *head2)
41 {
42     struct Node *tail1 = head1, *tail2 = head2;
43     while (tail1 && tail1->next)
44     {
45         tail1 = tail1->next;
46     }
47     while (tail2 && tail2->next)
48     {
49         tail2 = tail2->next;
50     }
51     struct Node *result = NULL;
52     int carry = 0;
53     while (tail1 || tail2 || carry)
54     {
55         int sum = carry;
56         if (tail1)
57         {
58             sum += tail1->digit;
59             tail1 = tail1->prev;
60         }
61         if (tail2)
62         {
63             sum += tail2->digit;
64             tail2 = tail2->prev;
65         }
66         append(&result, sum % 10);
67         carry = sum / 10;
68     }
69     struct Node *prev = NULL, *curr = result, *next = NULL;
70     while (curr)
71     {
72         next = curr->next;
73         curr->next = prev;
74         curr->prev = next;
75         prev = curr;
76         curr = next;
77     }
78     return prev;
79 }
80 int main()
81 {
82     char num1[101], num2[101];
83     printf("Enter first large integer: ");
84     scanf("%100s", num1);
85     printf("Enter second large integer: ");
86     scanf("%100s", num2);
87
88     struct Node *head1 = NULL, *head2 = NULL;
89     for (int i = 0; num1[i]; i++)
90     {
91         append(&head1, num1[i] - '0');
92     }
93     for (int i = 0; num2[i]; i++)
94     {
95         append(&head2, num2[i] - '0');
96     }
97
98     struct Node *result = addlists(head1, head2);
99
100    printf("Sum: ");
101    printList(result);
102
103    struct Node *temp;
104    while (head1)
105    {
106        temp = head1;
107        head1 = head1->next;
108        free(temp);
109    }
110    while (head2)
111    {
112        temp = head2;
113        head2 = head2->next;
114        free(temp);
115    }
116
117    return 0;
118 }
119
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa> cd "C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7"
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7> gcc lab7-1.c -o lab7-1 ; if ($?) { ./lab7-1 }
Enter a number: 19
The binary representation of 19 is: 10011
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7> cd "C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7"
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7> gcc lab7-2.c -o lab7-2 ; if ($?) { ./lab7-2 }
Enter first large integer: 21324132434321
Enter second large integer: 12423525342353
Sum: 33747657776674
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7>

```

Task 03: Circular Doubly Linked List:

(<https://github.com/varunnbn/dsa-sem3-iiitnr/blob/main/lab7/lab7-3.c>)

Use a circular doubly linked list to represent and add two polynomials.

Steps:

1. Each node contains: coefficient, power, and links (prev, next).

2. Insert terms of the two polynomials.

3. Traverse circularly and add like terms (same power).

4. Print the resulting polynomial in decreasing order of power.

Example:

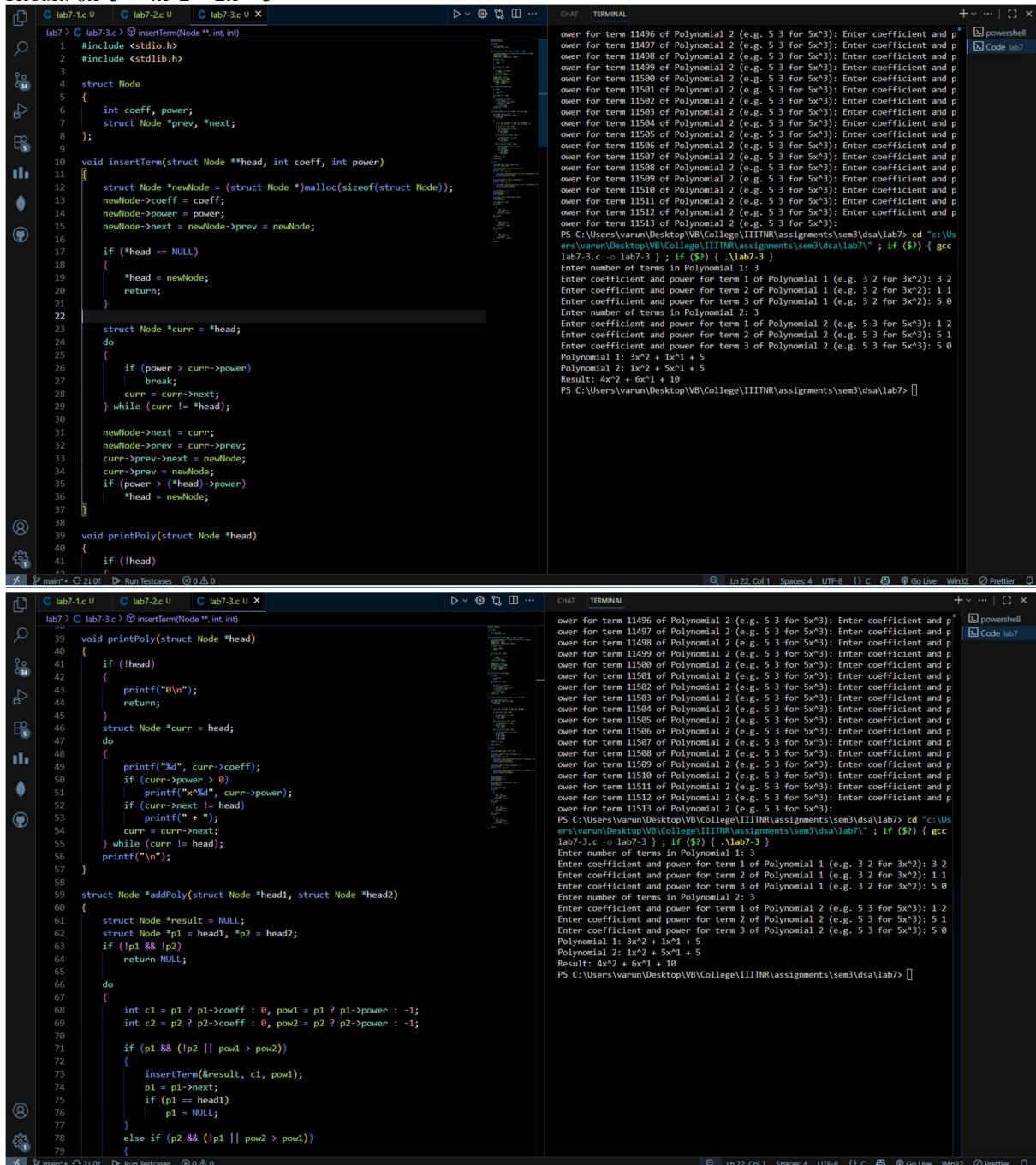
Polynomial A: $3x^3 + 4x^2 + 2$ (in CDLL 1)

Polynomial B: $5x^3 + 2x + 1$ (in CDLL 2)

Compare the power while traversing the list and add the sum in result list along

with power and display the final result.

Result: $8x^3 + 4x^2 + 2x + 3$



```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7> cd "c:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7"
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7> if ($?) { gcc lab7-3.c -o lab7-3 ; if ($?) { ./lab7-3 }
Enter number of terms in Polynomial 1: 3
Enter coefficient and power for term 1 of Polynomial 1 (e.g. 3 2 for 3x^2): 3 2
Enter coefficient and power for term 2 of Polynomial 1 (e.g. 3 2 for 3x^2): 1 1
Enter coefficient and power for term 3 of Polynomial 1 (e.g. 3 2 for 3x^2): 5 0
Enter number of terms in Polynomial 2: 3
Enter coefficient and power for term 1 of Polynomial 2 (e.g. 5 3 for 5x^3): 1 2
Enter coefficient and power for term 2 of Polynomial 2 (e.g. 5 3 for 5x^3): 5 1
Enter coefficient and power for term 3 of Polynomial 2 (e.g. 5 3 for 5x^3): 5 0
Polynomial 1: 3x^2 + 1x^1 + 5
Polynomial 2: 1x^2 + 5x^1 + 5
Result: 4x^2 + 6x^1 + 10
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7>
```

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7> cd "c:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7"
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7> if ($?) { gcc lab7-2.c -o lab7-2 ; if ($?) { ./lab7-2
Enter number of terms in Polynomial 1: 3
Enter coefficient and power for term 1 of Polynomial 1 (e.g. 3 2 for 3x^2): 3 2
Enter coefficient and power for term 2 of Polynomial 1 (e.g. 3 2 for 3x^2): 1 1
Enter coefficient and power for term 3 of Polynomial 1 (e.g. 3 2 for 3x^2): 5 0
Enter number of terms in Polynomial 2: 3
Enter coefficient and power for term 1 of Polynomial 2 (e.g. 5 3 for 5x^3): 1 2
Enter coefficient and power for term 2 of Polynomial 2 (e.g. 5 3 for 5x^3): 5 1
Enter coefficient and power for term 3 of Polynomial 2 (e.g. 5 3 for 5x^3): 5 0
Polynomial 1: 3x^2 + 1x^1 + 5
Polynomial 2: 1x^2 + 5x^1 + 5
Result: 4x^2 + 6x^1 + 10
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7>
```

```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7> cd "c:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7"
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7> if ($?) { gcc lab7-3.c -o lab7-3 ; if ($?) { ./lab7-3
Enter number of terms in Polynomial 1: 3
Enter coefficient and power for term 1 of Polynomial 1 (e.g. 3 2 for 3x^2): 3 2
Enter coefficient and power for term 2 of Polynomial 1 (e.g. 3 2 for 3x^2): 1 1
Enter coefficient and power for term 3 of Polynomial 1 (e.g. 3 2 for 3x^2): 5 0
Enter number of terms in Polynomial 2: 3
Enter coefficient and power for term 1 of Polynomial 2 (e.g. 5 3 for 5x^3): 1 2
Enter coefficient and power for term 2 of Polynomial 2 (e.g. 5 3 for 5x^3): 5 1
Enter coefficient and power for term 3 of Polynomial 2 (e.g. 5 3 for 5x^3): 5 0
Polynomial 1: 3x^2 + 1x^1 + 5
Polynomial 2: 1x^2 + 5x^1 + 5
Result: 4x^2 + 6x^1 + 10
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7>
```

CHAT TERMINAL

```

ower for term 11496 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11497 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11498 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11499 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11500 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11502 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11504 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11505 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11506 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11508 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11510 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11511 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11513 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
PS C:\Users\varun\Desktop\VB\College\IIIITNR\assignments\sem3\dsa\lab7> cd "c:\Users\varun\Desktop\VB\College\IIIITNR\assignments\sem3\dsa\lab7" ; if ($?) { gcc lab7-3.c -o lab7-3 } ; if ($?) { lab7-3 }
Enter number of terms in Polynomial 1: 3
Enter coefficient and power for term 1 of Polynomial 1 (e.g. 3 2 for 3x^2): 3 2
Enter coefficient and power for term 2 of Polynomial 1 (e.g. 3 2 for 3x^2): 1 1
Enter coefficient and power for term 3 of Polynomial 1 (e.g. 3 2 for 3x^2): 5 0
Enter number of terms in Polynomial 2: 3
Enter coefficient and power for term 1 of Polynomial 2 (e.g. 5 3 for 5x^3): 1 2
Enter coefficient and power for term 2 of Polynomial 2 (e.g. 5 3 for 5x^3): 5 1
Enter coefficient and power for term 3 of Polynomial 2 (e.g. 5 3 for 5x^3): 5 0
Polynomial 1: 3x^2 + 6x^1 + 5
Polynomial 2: 1x^2 + 5x^1 + 5
Result: 4x^2 + 6x^1 + 10
PS C:\Users\varun\Desktop\VB\College\IIIITNR\assignments\sem3\dsa\lab7>

```

CHAT TERMINAL

```

ower for term 11496 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11497 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11498 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11499 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11500 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11502 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11504 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11505 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11506 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11508 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11510 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11511 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
ower for term 11513 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and p*
PS C:\Users\varun\Desktop\VB\College\IIIITNR\assignments\sem3\dsa\lab7> cd "c:\Users\varun\Desktop\VB\College\IIIITNR\assignments\sem3\dsa\lab7" ; if ($?) { gcc lab7-3.c -o lab7-3 } ; if ($?) { lab7-3 }
Enter number of terms in Polynomial 1: 3
Enter coefficient and power for term 1 of Polynomial 1 (e.g. 3 2 for 3x^2): 3 2
Enter coefficient and power for term 2 of Polynomial 1 (e.g. 3 2 for 3x^2): 1 1
Enter coefficient and power for term 3 of Polynomial 1 (e.g. 3 2 for 3x^2): 5 0
Enter number of terms in Polynomial 2: 3
Enter coefficient and power for term 1 of Polynomial 2 (e.g. 5 3 for 5x^3): 1 2
Enter coefficient and power for term 2 of Polynomial 2 (e.g. 5 3 for 5x^3): 5 1
Enter coefficient and power for term 3 of Polynomial 2 (e.g. 5 3 for 5x^3): 5 0
Polynomial 1: 3x^2 + 6x^1 + 5
Polynomial 2: 1x^2 + 5x^1 + 5
Result: 4x^2 + 6x^1 + 10
PS C:\Users\varun\Desktop\VB\College\IIIITNR\assignments\sem3\dsa\lab7>

```

```
lab7 > C lab7-1.c U C lab7-2.c U C lab7-3.c U
lab7 > C lab7-3.c > insertTerm(Node **, int, int)
100  int main()
134      if (curr)
135          do
136              {
137                  temp = curr;
138                  curr = curr->next;
139                  free(temp);
140              } while (curr != poly1);
142      }
143      curr = poly2;
144      if (curr)
145      {
146          do
147          {
148              temp = curr;
149              curr = curr->next;
150              free(temp);
151          } while (curr != poly2);
152      }
153      curr = result;
154      if (curr)
155      {
156          do
157          {
158              temp = curr;
159              curr = curr->next;
160              free(temp);
161          } while (curr != result);
162      }
163
164  return 0;
165 }

CHAT TERMINAL
owner for term 11496 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11496 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11497 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11498 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11499 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11500 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11501 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11502 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11503 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11504 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11505 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11506 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11507 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11508 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11509 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11510 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11511 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11512 of Polynomial 2 (e.g. 5 3 for 5x^3); Enter coefficient and power for term 11513 of Polynomial 2 (e.g. 5 3 for 5x^3);
PS C:\Users\varun\Desktop\VB\College\IITINR\assignments\sem3\dsa\lab7> cd "C:\Users\varun\Desktop\VB\College\IITINR\assignments\sem3\dsa\lab7" ; if ($?) { gcc lab7-3.c -o lab7-3 } ; if ($?) { ./lab7-3 }
Enter number of terms in Polynomial 1: 3
Enter coefficient and power for term 1 of Polynomial 1 (e.g. 3 2 for 3x^2): 3 2
Enter coefficient and power for term 2 of Polynomial 1 (e.g. 3 2 for 3x^2): 1 1
Enter coefficient and power for term 3 of Polynomial 1 (e.g. 3 2 for 3x^2): 5 0
Enter number of terms in Polynomial 2: 3
Enter coefficient and power for term 1 of Polynomial 2 (e.g. 5 3 for 5x^3): 1 2
Enter coefficient and power for term 2 of Polynomial 2 (e.g. 5 3 for 5x^3): 5 1
Enter coefficient and power for term 3 of Polynomial 2 (e.g. 5 3 for 5x^3): 5 0
Polynomial 1: 3x^2 + 1x^1 + 5
Polynomial 2: 1x^2 + 5x^1 + 5
Result: 4x^2 + 6x^1 + 10
PS C:\Users\varun\Desktop\VB\College\IITINR\assignments\sem3\dsa\lab7>
```