VARUN BHARGAVA – 241010282
DATA STRUCTURES TASK-7

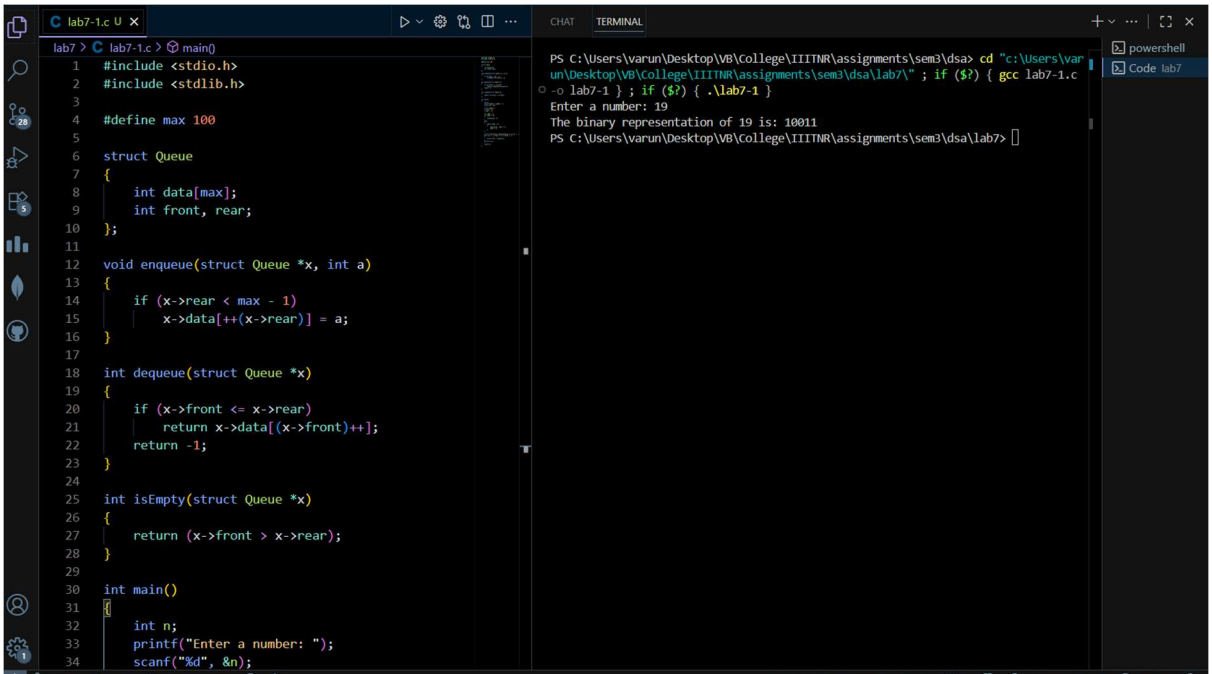
Task 01: Queue:

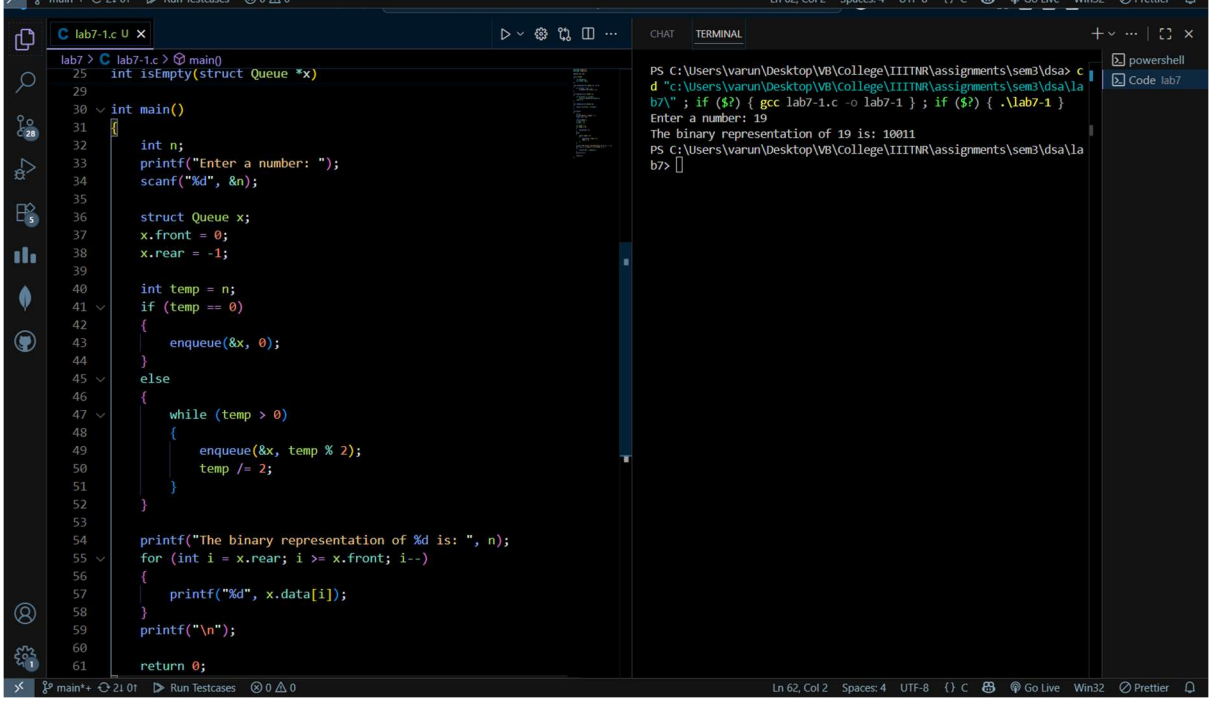
(https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab7/lab7-1.c)

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.

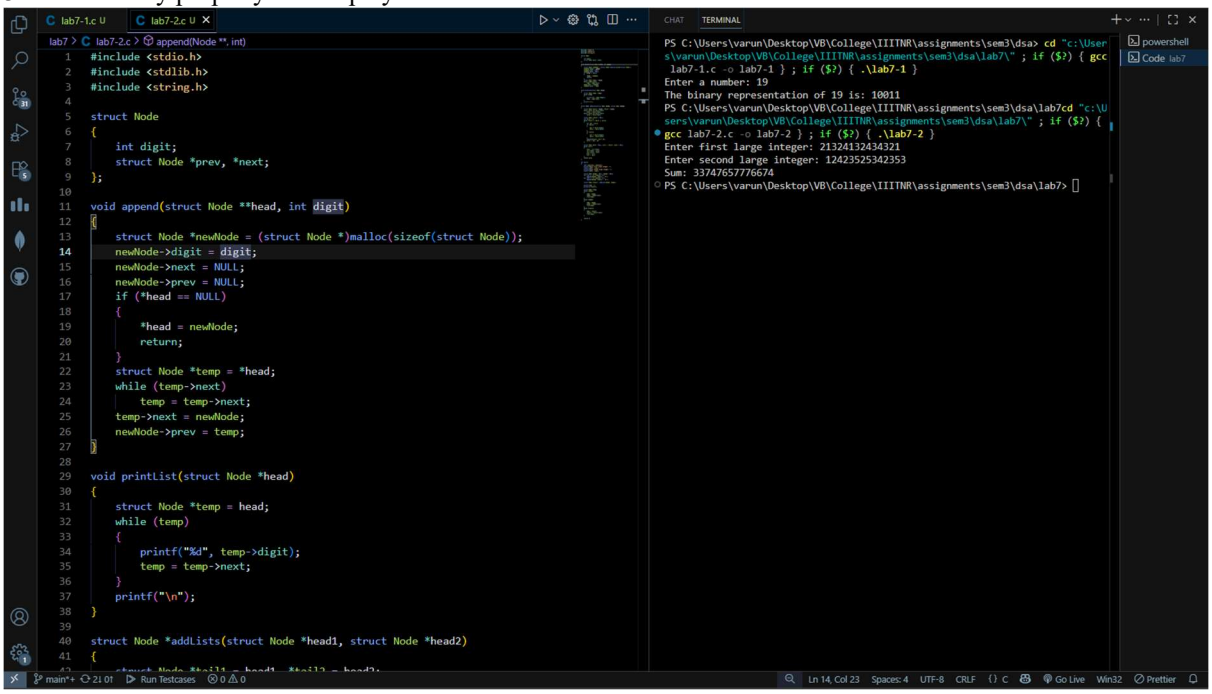


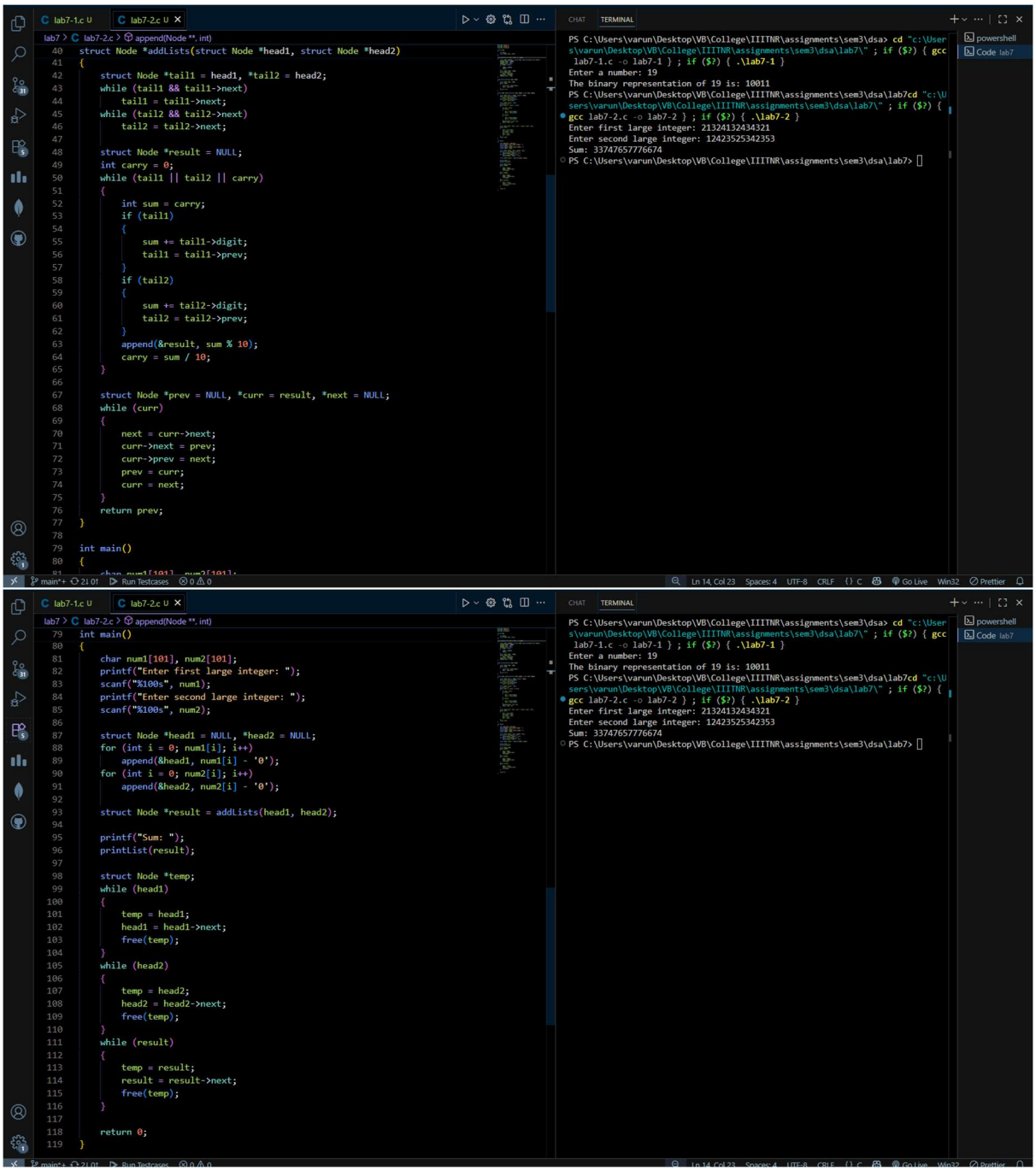


Task 02: Doubly Linked List:

(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:
3. 1. Store each digit of the two numbers in separate doubly linked lists.
4. 2. Perform addition digit by digit from the rightmost (using backward traversal).
5. 3. Handle carry properly and display the result list.





```c
struct Node *addLists(struct Node *head1, struct Node *head2)
{
    struct Node *tail1 = head1, *tail2 = head2;
    while (tail1 && tail1->next)
        tail1 = tail1->next;
    while (tail2 && tail2->next)
        tail2 = tail2->next;

    struct Node *result = NULL;
    int carry = 0;
    while (tail1 || tail2 || carry)
    {
        int sum = carry;
        if (tail1)
        {
            sum += tail1->digit;
            tail1 = tail1->prev;
        }
        if (tail2)
        {
            sum += tail2->digit;
            tail2 = tail2->prev;
        }
        append(&result, sum % 10);
        carry = sum / 10;
    }

    struct Node *prev = NULL, *curr = result, *next = NULL;
    while (curr)
    {
        next = curr->next;
        curr->next = prev;
        curr->prev = next;
        prev = curr;
        curr = next;
    }
    return prev;
}

int main()
{
```

```
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\lab7cd "c:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7\" ; if ($?) { gcc lab7-2.c -o lab7-2 } ; if ($?) { .\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>
```



```c
int main()
{
    char num1[101], num2[101];
    printf("Enter first large integer: ");
    scanf("%100s", num1);
    printf("Enter second large integer: ");
    scanf("%100s", num2);

    struct Node *head1 = NULL, *head2 = NULL;
    for (int i = 0; num1[i]; i++)
        append(&head1, num1[i] - '0');
    for (int i = 0; num2[i]; i++)
        append(&head2, num2[i] - '0');

    struct Node *result = addLists(head1, head2);

    printf("Sum: ");
    printList(result);

    struct Node *temp;
    while (head1)
    {
        temp = head1;
        head1 = head1->next;
        free(temp);
    }
    while (head2)
    {
        temp = head2;
        head2 = head2->next;
        free(temp);
    }
    while (result)
    {
        temp = result;
        result = result->next;
        free(temp);
    }

    return 0;
}
```

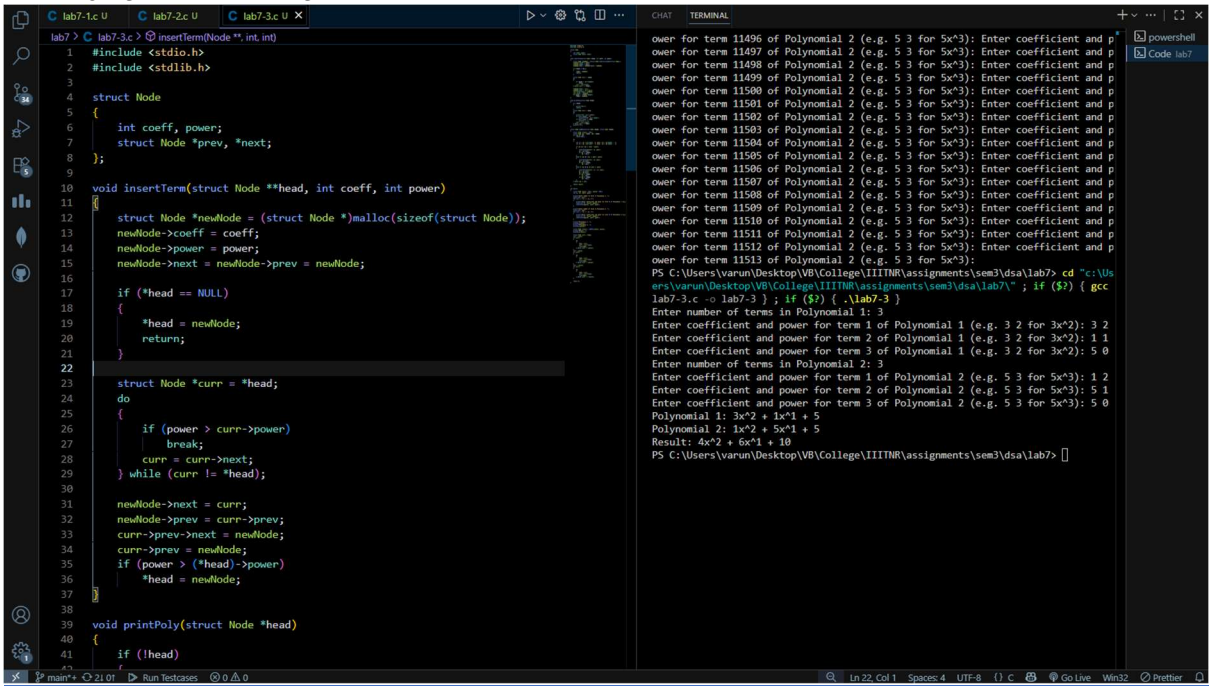
```
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa> cd "c:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7\" ; if ($?) { gcc lab7-1.c -o lab7-1 } ; if ($?) { .\lab7-1 }
Enter a number: 19
The binary representation of 19 is: 10011
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7cd "c:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7\" ; if ($?) { gcc lab7-2.c -o lab7-2 } ; if ($?) { .\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>
```

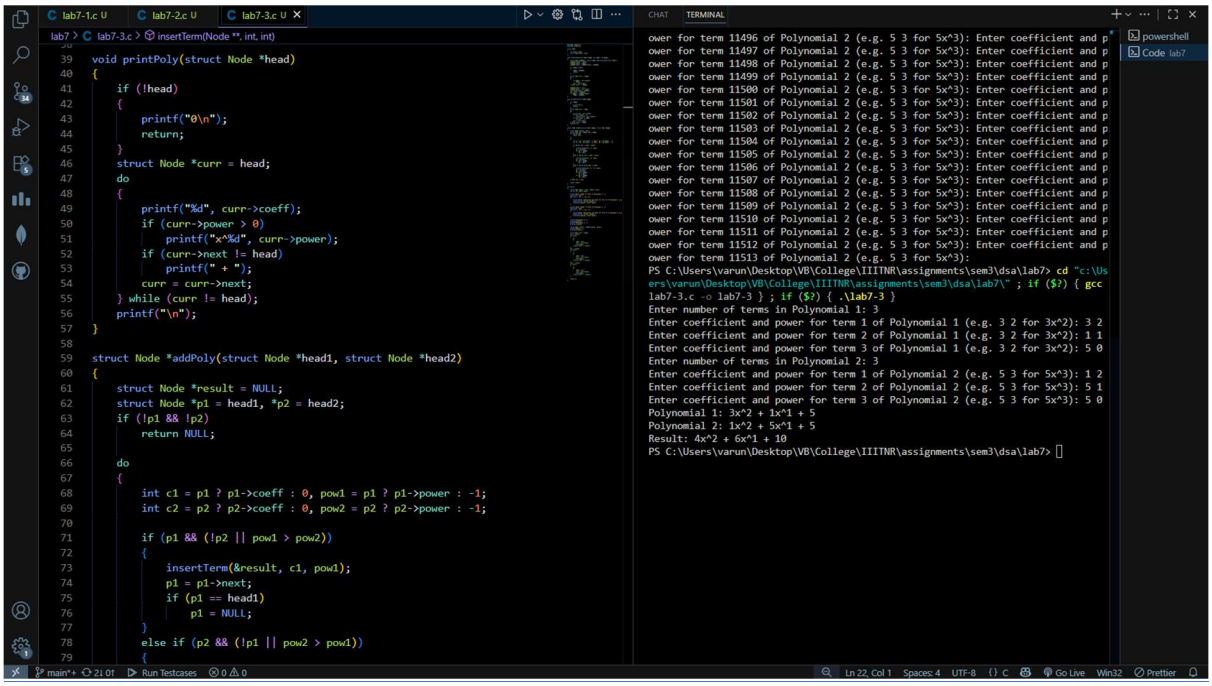
Task 03: Circular Doubly Linked List:

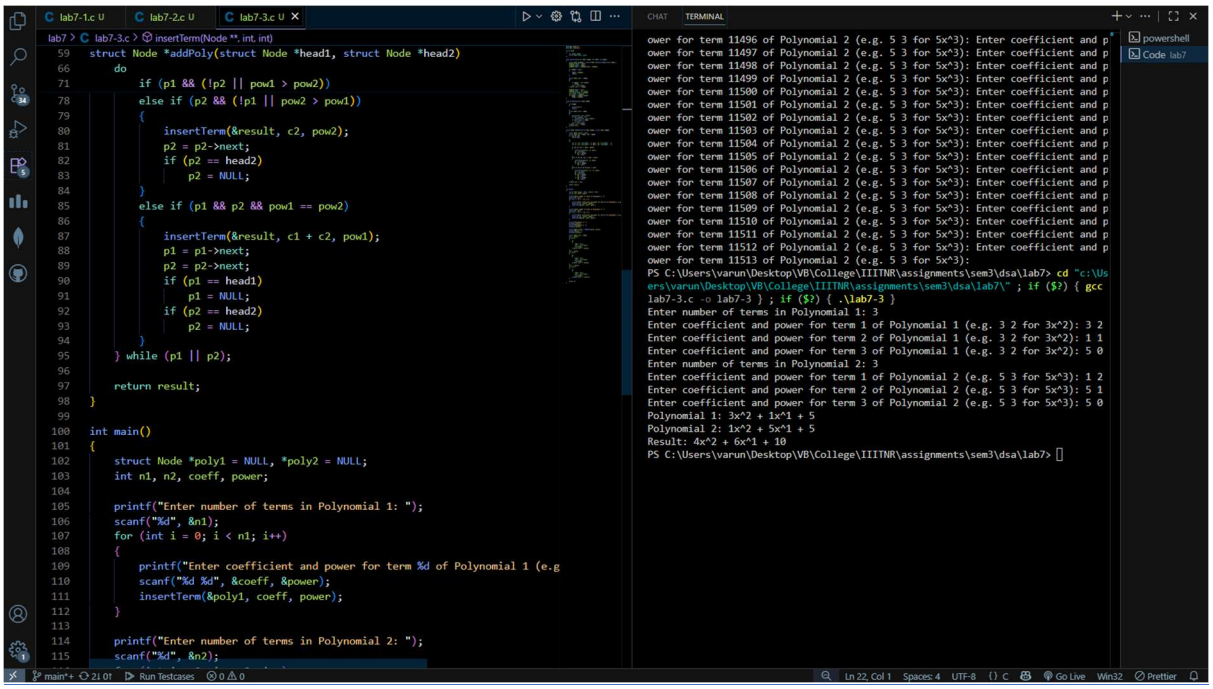
(https://github.com/varunnnb/dsa-sem3-iiitnr/blob/main/lab7/lab7-3.c)
Use a circular doubly linked list to represent and add two polynomials.
Steps:
1. Each node contains: coefficient, power, and links (prev, next).

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

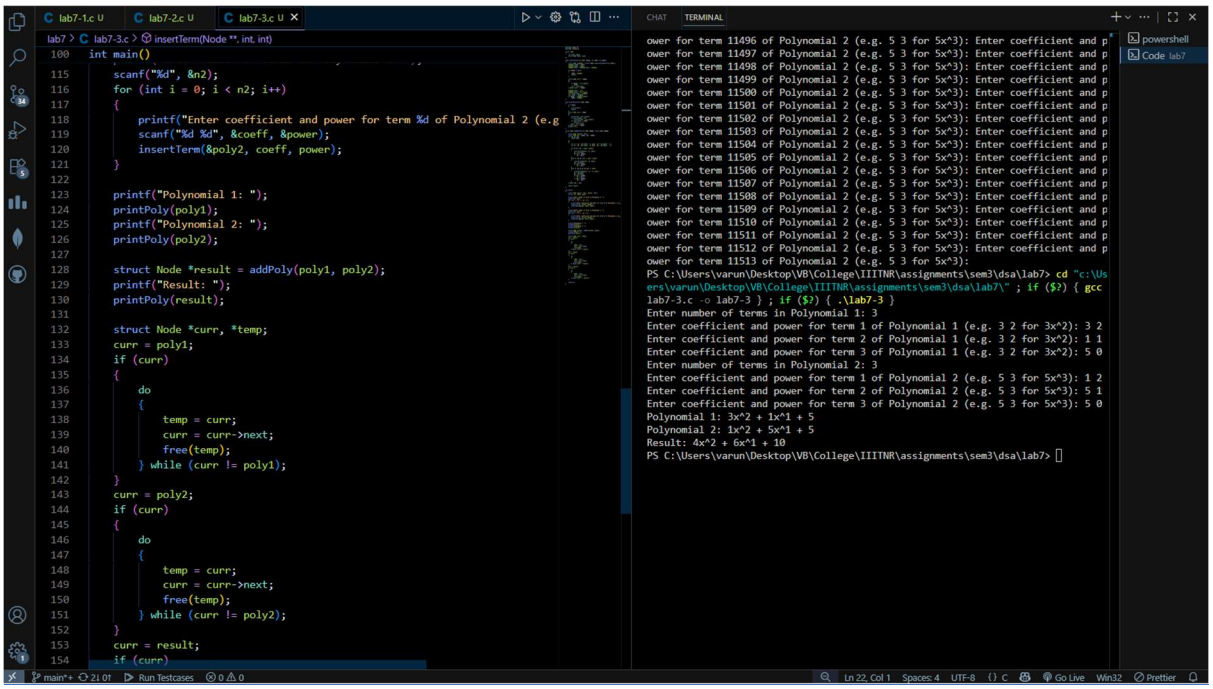






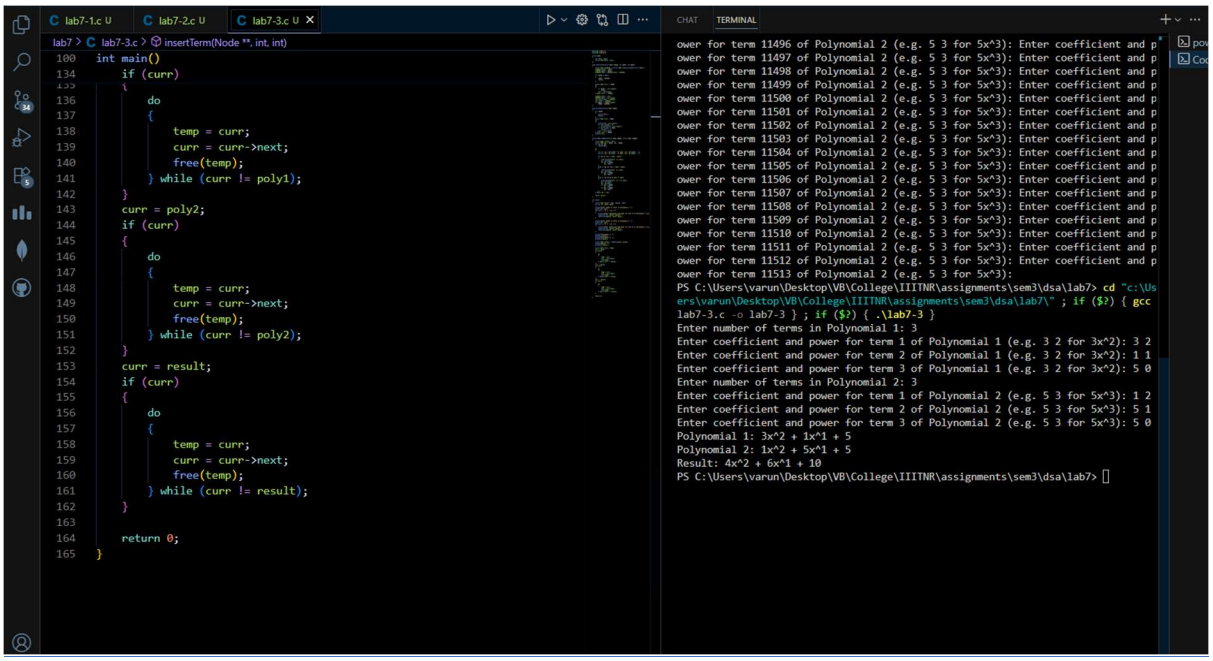
```
struct Node *addPoly(struct Node *head1, struct Node *head2)
    do
        if (p1 && (!p2 || pow1 > pow2))
        else if (p2 && (!p1 || pow2 > pow1))
        {
            insertTerm(&result, c2, pow2);
            p2 = p2->next;
            if (p2 == head2)
                p2 = NULL;
        }
        else if (p1 && p2 && pow1 == pow2)
        {
            insertTerm(&result, c1 + c2, pow1);
            p1 = p1->next;
            p2 = p2->next;
            if (p1 == head1)
                p1 = NULL;
            if (p2 == head2)
                p2 = NULL;
        }
    } while (p1 || p2);

    return result;
}

int main()
{
    struct Node *poly1 = NULL, *poly2 = NULL;
    int n1, n2, coeff, power;

    printf("Enter number of terms in Polynomial 1: ");
    scanf("%d", &n1);
    for (int i = 0; i < n1; i++)
    {
        printf("Enter coefficient and power for term %d of Polynomial 1 (e.g
        scanf("%d %d", &coeff, &power);
        insertTerm(&poly1, coeff, power);
    }

    printf("Enter number of terms in Polynomial 2: ");
    scanf("%d", &n2);
```

```
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 11501 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 11503 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 11507 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 11509 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 11512 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):
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7> cd "c:\Us
ers\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>
```



```
int main()
    scanf("%d", &n2);
    for (int i = 0; i < n2; i++)
    {
        printf("Enter coefficient and power for term %d of Polynomial 2 (e.g
        scanf("%d %d", &coeff, &power);
        insertTerm(&poly2, coeff, power);
    }

    printf("Polynomial 1: ");
    printPoly(poly1);
    printf("Polynomial 2: ");
    printPoly(poly2);

    struct Node *result = addPoly(poly1, poly2);
    printf("Result: ");
    printPoly(result);

    struct Node *curr, *temp;
    curr = poly1;
    if (curr)
    {
        do
        {
            temp = curr;
            curr = curr->next;
            free(temp);
        } while (curr != poly1);
    }
    curr = poly2;
    if (curr)
    {
        do
        {
            temp = curr;
            curr = curr->next;
            free(temp);
        } while (curr != poly2);
    }
    curr = result;
    if (curr)
```

```
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 11501 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 11503 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 11507 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 11509 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 11512 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):
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7> cd "c:\Us
ers\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>
```


lab7-1.c U
lab7-2.c U
lab7-3.c U
lab7 > lab7-3.c > insertTerm(Node **, int, int)
int main()
if (curr)
{
do
{
temp = curr;
curr = curr->next;
free(temp);
} while (curr != poly1);
}
curr = poly2;
if (curr)
{
do
{
temp = curr;
curr = curr->next;
free(temp);
} while (curr != poly2);
}
curr = result;
if (curr)
{
do
{
temp = curr;
curr = curr->next;
free(temp);
} while (curr != result);
}
return 0;
}
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 11501 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 11503 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 11507 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 11509 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 11512 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):
PS C:\Users\varun\Desktop\VB\College\IIITNR\assignments\sem3\dsa\lab7> cd "c:\Us
ers\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>