

LINKED LIST

```
#include <stdio.h>
#include <stdlib.h>
struct Node { int data ; struct node* next ; } head = null;
```

```
Void insert AT (int data , int pos) {
```

```
    Struct nod * New node = malloc (size of  
    (Struct nod) );
```

```
    Newnode → data = data ;
```

```
    Newnode = NULL ;
```

```
    if (pos == 1 || b head) {
```

```
        Newnode → Next = head ;
```

```
        head = Newnode ;
```

```
        return ;
```

```
}
```

```
    Struct nod * temp = head ;
```

```
    newnode → next = temp → next ;
```

```
    temp → next = newnode ;
```

```
}
```

```
void printlist () {
```

```
    Struct Node *temp = head ;
```

```
    while (temp) {
```

```
        printf ("%d → ", temp → data ) ;
```

```
        temp = temp → next )
```

```
    }
```

```
    printf ("Null \n") ;
```

EXPT. NO.	NAME	M	T	W	T	F	S	S
		Page No.:	YOUVA					

```

int main () {
    int choice, data, pos;
    while (1) {
        printf (" 1 Insert 2Delete 3pos 4print 5exit \n choose:");
        scanf ("%d", &choice);
        if (choice == 5) break;
        if (choice == 2) pos = 10000;
        else if (choice == 3) {
            printf (" Data pos: ");
            scanf ("%d %d", &data, &pos);
            insert AT (data, pos);
            continue;
        }
        printf ("Data: ");
        scanf ("%d", &data);
        insert AT (data, choice == 1 ? 1 : pos);
        if (choice == 4) printlist ();
    }
    return 0;
}

```

C arith.c

C question.c X

```
6 void insertAt(int data, int pos) {
10    if (pos == 1 || !head) {
13        return;
14    }
15    struct Node* temp = head;
16    for (int i = 1; i < pos - 1 && temp
17        newNode->next = temp->next;
18        temp->next = newNode;
19    }
20
21    void printList() {
22        struct Node* temp = head;
23        while (temp) {
24            printf("%d -> ", temp->data);
25            temp = temp->next;
26        }
27        printf("NULL\n");
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORT

Choose: 1

Data: 12

1-Beg 2-End 3-Pos 4-Print 5-Exit

Choose: 1

Data: 12

1-Beg 2-End 3-Pos 4-Print 5-Exit

Choose: 2

Data: 23

1-Beg 2-End 3-Pos 4-Print 5-Exit

Choose: 3

Data pos: 12

4

1-Beg 2-End 3-Pos 4-Print 5-Exit

Choose: 4

Data: 12

12 -> 12 -> 23 -> 12 -> 12 -> NULL

1-Beg 2-End 3-Pos 4-Print 5-Exit

Choose:



EXPT. NO.	NAME	M	T	W	T	F	S	S
		Page No.:	YOUVA					

12 - 1 Preprocessors & directives

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

```
#
```

```
#define PI 3.14
```

```
int main () {
```

```
    printf ("value of PI %.2f \n", PI);
```

```
    return 0;
```

```
}
```

C exp12-1.c X

```
C exp12-1.c > ...
1 #include <stdio.h>
2
3 #define PI 3.14 // Defining a constant using
4
5 int main() {
6     printf("Value of PI: %.2f\n", PI);
7     return 0;
8 }
9
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Yash\EXP> gcc exp12-1.c
PS C:\Users\Yash\EXP> ./a.exe
Value of PI: 3.14
PS C:\Users\Yash\EXP>
```



EXPT. NO.	NAME	M	T	W	T	F	S	S
		Page No.:					YOUVA	

Exp 12-2

#include <stdio.h>

#define SQUARE(n) ((n)*(n))

int main () {

int num = 5;

printf ("Square of %d is %d\n", num, SQUARE(num));

return 0;

}

C exp12-2.c > ..

```
1 #include <stdio.h>
2
3 // Function-like macro to calculate square
4 #define SQUARE(x) ((x) * (x))
5
6 int main() {
7     int num = 5;
8     printf("Square of %d is %d\n",
9            num, SQUARE(num));
10 }
11
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PS C:\Users\Yash\EXP> gcc exp12-1.c

PS C:\Users\Yash\EXP> ./a.exe

Value of PI: 3.14

PS C:\Users\Yash\EXP> gcc exp12-2.c

PS C:\Users\Yash\EXP> ./a.exe

Square of 5 is 25

PS C:\Users\Yash\EXP>

4/4



EXPT. NO.	NAME	M T W T F S S
		Page No.: Date: YOUVA

Exp 13-1 Macros

include <stdio.h>

define ADD(a,b) ((a)+(b))

define SUBTRACT(a,b) ((a)-(b))

define MULTIPLY (a,b) ((a)*(b))

#

define DIVIDE (a,b) ((a)/(b))

int main () {

int a = 10 , b=5;

printf ("Addition of a & b = %.d \n", ADD(a,b));

printf ("Subtraction of a & b = %.d \n", ~~Subtract(a,b)~~);

printf ("Multiplication of a & b = %.d \n", Multiply (a,b));

printf ("Division of a & b = %.d \n", Divide (a,b));

} return 0;

3

C exp13-1.c X

```

C exp13-1.c > ...
1 #include <stdio.h>
2
3 // Macro definitions for arithmetic operations
4 #define ADD(a, b) ((a) + (b))
5 #define SUBTRACT(a, b) ((a) - (b))
6 #define MULTIPLY(a, b) ((a) * (b))
7 #define DIVIDE(a, b) ((b) != 0 ? (a) / (b) : 0)
8
9 int main() {
10     int x = 10, y = 5;
11
12     printf("Addition: %d + %d = %d\n", x, y, ADD(x, y));
13     printf("Subtraction: %d - %d = %d\n", x, y, SUBTRACT(x, y));
14     printf("Multiplication: %d * %d = %d\n", x, y, MULTIPLY(x, y));
15     printf("Division: %d / %d = %d\n", x, y, DIVIDE(x, y));
16
17     return 0;
18 }
19

```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

```

PS C:\Users\Yash\EXP> gcc exp12-2.c
PS C:\Users\Yash\EXP> ./a.exe
Square of 5 is 25
PS C:\Users\Yash\EXP> gcc exp13-1.c
PS C:\Users\Yash\EXP> ./a.exe
Addition: 10 + 5 = 15
Subtraction: 10 - 5 = 5
Multiplication: 10 * 5 = 50
Division: 10 / 5 = 2
PS C:\Users\Yash\EXP>

```



Exp 14-1 Static Library

arithm.c

```
int add (int a, int b) {  
    return (a+b);  
}  
  
int sub (int a, int b) {  
    return (a-b);  
}  
  
int multiply (int a, int b) {  
    return (a*b);  
}  
  
int divide (int a, int b) {  
    if (b==0) return 0;  
    else return a/b;  
}
```

exp

header - c

```
#ifndef Arithm_H  
#define Arithm_H
```

```
int add (int a, int b);  
int sub (int a, int b);  
int multiply (int a, int b);  
int divide (int a, int b);
```

#endif

EXPT. NO.	NAME	M T W T F S S
		Page No.:
		Date: YOUVA

exp 14-2

```
#include <stdio.h>
#include "arith.h"
```

```
int main () {
    int n = 20, y = 4;
```

```
printf ("add : %.d \n", add (n,y));
printf ("sub : %.d \n", sub (n,y));
printf ("multiplication : %.d \n", multiply (n,y));
printf ("divide : %.d \n", divide (n,y));
```

```
return 0;
```

3

C arith.c

C main.c



```
C main.c> ...
1 #include <stdio.h>
2 #include "arith.h"
3
4 int main() {
5     int x = 20, y = 4;
6
7     printf("Add: %d\n", add(x, y));
8     printf("Subtract: %d\n", subtract(x, y));
9     printf("Multiply: %d\n", multiply(x, y));
10    printf("Divide: %d\n", divide(x, y));
11
12    return 0;
13 }
14
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
>>
ith
>> ./test_arith
>>
Add: 24
Subtract: 16
Multiply: 80
Divide: 5
PS C:\Users\Yash\EXP>
PS C:\Users\Yash\EXP>
```

