

OUTPUT-1.

Run 1 :

Enter the number of elements in the array : 5

Enter the array Elements

7 8 5 6 3

Enter the element to be searched for : 6

Element found at position 4

Run 2 :

Enter the number of Elements in the array : 5

Enter the array Elements

7 8 5 6 3

Enter the Element to be searched for : 10

Element not found

Q-1 Write a program to search an element from array using linear search and its print location.

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

```
void main()
```

```
{
```

```
int a[40], n, i, se, found = 0;
```

```
printf ("Enter the number of elements in the array:");
```

```
scanf ("%d", &n);
```

```
printf ("Enter the array elements (n):");
```

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

```
scanf ("%d", &a[i]);
```

```
printf ("Enter the element to be searched for:");
```

```
scanf ("%d", &se);
```

```
i = 1
```

```
while (i <= n & & found == 0)
```

```
{
```

```
if (a[i] == se) found = 1;
```

```
i++;
```

```
}
```

```
/* write output */
```

```
if (found == 1)
```

```
printf ("Element found at position: %d", i);
```

```
else
```

```
printf ("Element not found");
```

```
?
```

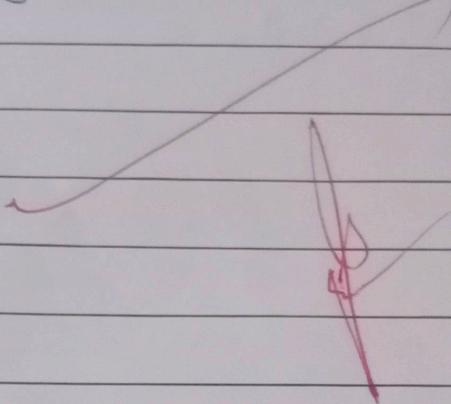
Q2) Write a program which search an element from array using Binary search and print its location.

```
#include <stdio.h>
main()
{
    int a[40], n, i, se, first, last, mid;
    printf("Enter the number of elements in the array:");
    scanf("%d", &n);
    printf("Enter the array elements (separated)\n");
    for (i = 1; i < n; ++i)
        scanf("%d", &a[i]);
    printf("Enter the elements to be searched for:");
    scanf("%d", &se);
    /* initialise the variables */
    first = 1;
    last = n;
    mid = (first + last) / 2;
    /* search for array elements */
    while (first < last && a[mid] != se)
    {
        if (a[mid] < se) first = mid + 1;
        if (a[mid] > se) last = mid - 1;
        mid = (first + last) / 2;
    }
    /* write output */
    if (a[mid] == se)
```

print("Element found at position: l..d", mid);
else

3.

print("Element not found");



Q3 Write a program to insert element in array at the desired location.

```
#include <stdio.h>
#include <conio.h>
main()
{
    int a[20], m, i, j, K, new_element;
    clrscr();
    printf("Enter the number of elements:");
    scanf("%d", &m);
    printf("Enter %d elements\n", m);
    for (K = 1; K <= m; K++)
        scanf("%d", &a[K]);
    printf("Enter the new element to be inserted:");
    scanf("%d", &new_element);
    printf("Where %d is to be inserted:", new_element);
    scanf("%d", &j);
    i = m
    while (i >= j)
    {
        a[i + 1] = a[i];
        i++;
    }
    a[j] = new_element;
    m = m + 1;
    printf("Array after insertion is\n");
    for (K = 1; K <= m; K++)
        printf("%d ", a[K]);
}
```

```
printf ("%d\n", a[k]);  
return 0;
```

3.

Q-4 Write a program to delete element in array
at the desired location.

```
#include <stdio.h>  
#include <conio.h>  
main()  
{  
    int a[20], m, i, j, k, del;  
    clrscr();  
    printf ("Enter the no. of elements: ");  
    scanf ("%d", &m);  
    printf ("Enter %d elements\n", m);  
    for (k = 1; k <= m; k++)  
        scanf ("%d", &a[k]);  
    del = a[j];
```

```
    printf ("Enter the position of element to  
    deleted");  
    scanf ("%d", &j);
```

```
i = j;  
while (i <= m - 1)
```

4.

```
a[i] = a[i + 1];  
i++;
```

3

$m = m - 1;$

`printf ("Array after deletion is\n");`

`for (k=1; k <= m; k++)`

`printf ("%d\n", a[k]);`

`return (0);`

2

B
F

Q-5 Write a program all arithmetic operations using two dimensional array.

```
#include <stdio.h>
int main()
{
    int i, j, row, col, A[20][20], B[20][20];
    int add[10][10], sub[10][10], mult[10][10], mat[10][10];
    float div[10][10];
    printf("Enter no. of rows and columns:");
    scanf("%d %d", &i, &j);
    printf("Enter elements of 1st array:");
    for (row = 0; row < i; row++)
    {
        for (col = 0; col < j; col++)
        {
            scanf("%d", &A[row][col]);
        }
    }
    printf("Enter element of second array:");
    for (row = 0; row < i; row++)
    {
        for (col = 0; col < j; col++)
        {
            scanf("%d", &B[row][col]);
        }
    }
```

add [row][col] = A[row][col] + B[row][col];
sub [row][col] = A[row][col] - B[row][col];
Mul [row][col] = A[row][col] * B[row][col];
div [row][col] = A[row][col] / B[row][col];
mod [row][col] = A[row][col] % B[row][col];

}

}

printf ("Add It sub It Mult It Div It Mod");
printf.

for (col = 0, col < j; col++).

{

printf ("1.d1", add [row][col]);
printf ("1.d2", sub [row][col]);
printf ("1.d3", mult [row][col]);
printf ("1.d4", div [row][col]);
printf ("1.d5", mod [row][col]);

}

getche();

return 0;

}

B. Write a program in C to insert a node at the front and end of linked list

```
#include <stdio.h>
#include <conio.h>
struct node
{
    int data;
    struct node *next;
    *head;
}
void initialize()
{
    head = NULL
}
void insert_at_front(int num)
{
    struct node *new_node = (struct node *)malloc(
        sizeof(struct node));
    new_node->data = num;
    new_node->next = head;
    head = new_node;
    printf("Inserted Elements : %d\n", num);
}
void insert_at_end(struct node *head, int num)
{
    if (head == NULL)
```

f

```
printf ("Enter Error: Invalid node pointer!!!\n");  
return;
```

}

```
struct node * new_node = ( struct node ) malloc ( sizeof  
( struct node ) );
```

new_node -> data = num;

new_node -> next = NULL;

while (head -> next != NULL).

head = head -> next;

head -> next = new_node;

3

```
void print_linked_list ( struct node * node ( * ) ).
```

E

```
printf ("In linked list\n");
```

while (node . ite != NULL).

E

```
printf ("-%d", node . ite -> data);
```

node . ite = node . ite -> next;

if (node . ite != NULL)

```
printf ("--> ");
```

3

```
int main ( )
```

E

```
initialize();
```

```
insert_at_Front ( 2 );
```

```
insert_at_Front ( 4 );
```

```
insert_at_Front ( 5 );
```

```
insert At Front (a);  
printf 'linked list (head);  
insert At End (head; (o);  
printf ("In In After insertion At End \n");  
printf ' linked - list (head);  
return 0;
```

?

Write a program to find Substring from the entered string.

```
#include <stdio.h>
int main()
{
    char str1[80], str2[80];
    int j, i, f;
    printf("Enter first string : ");
    gets(str1);
    printf("Enter second string : ");
    gets(str2);
    for (j = 0; str2[j] != '\0'; j++)
        for (i = 0; f == 0; str1[i] = str2[j], str2[i] = '0', j++)
            if (str1[i] == str2[j])
                j++;
            else
                f = 1;
            if (f == 1)
                printf("Substring found at position %d", j + 1);
        else
            f = 0;
}
```

Date	/	/
Page No.	24	

printf ("Substring not found ");

3.

return 0;

3.

Q. Write a program in C to search an element from the linked list.

```
#include <stdio.h>
#include <conio.h>
struct node {
    int data;
    struct node* next;
}
first * new
int search (int item)
{
    int count = 1;
    new = & first;
    while (new->next != NULL) {
        if (new->data == item)
            break;
        else
            count++;
        new = new->next;
    }
    int main()
{
    int no, i, item, pos;
    first.next = NULL;
    new = & first;
    printf ("How many nodes you want in linked list ");
    scanf ("%d", &no);
    printf ("\n");
}
```

for (i=0; i<n; i++) {

 no->next = (Struct node *) malloc(sizeof(Struct node));

 printf("Enter element in node %d.d ", i+1);

 scanf("%d", &(new->data));

 new = new->next;

}

 new->next = null;

 printf("Enter element in linked list (%d)",

 new = &first;

 while .

 printf("%d.d ", i, new->data);

 new = new->next;

}

 printf("\n");

 printf("Enter Element to be searched");

 scanf("%d", &iitem);

 Pos = search(first, &iitem);

 if (Pos <= n)

 printf("%d.d is found at node = %d.d ", item Pos);

 Else {

 printf("Sorry ! %d.d is not in linked list ", item);

 return 0;

}

Q. Write a program to update an element from the linked list.

```
#include < stdio.h>
#include < std:: lab.h>
struct node {
    int data;
    struct node* next;
};
```

```
void update_element(struct node* head, int position, int
new value);
```

{

```
struct node* current = head;
int current position = 0;
```

```
while (current != NULL && current position < position)
{
```

```
    current = current → next;
```

```
    current position ++;
```

{

```
If (current = NULL)
```

```
    printf ("In valid position\n");
```

```
    return;
```

{

```
    current → data = new value;
```

{

```
void Printlist (struct Node* head) {
```

```
    struct node * current = head;
```

```
struct node * current = head;
while (current != NULL) {
    printf(" -> ", current->data);
    current = current->next;
}
printf(" Null \n");
int main() {
    struct node * head = NULL;
    struct node * second = NULL;
    struct node * third = NULL;
    struct node * fourth = NULL;
    head = (struct node *) malloc(sizeof(struct node));
    second = (struct node *) malloc(sizeof(struct node));
    third = (struct node *) malloc(sizeof(struct node));
    fourth = (struct node *) malloc(sizeof(struct node));
    head->data = 1;
    head->next = second;
    second->data = 2;
    second->next = third;
    third->data = 3;
    third->next = fourth;
    fourth->data = 4;
    fourth->next = NULL;
    printf("Original linked list: ");
    printList(head);
    updateElement(head, 2, 10);
    printf("Updated linked list: ");
    printList(head);
    return 0;
}
```

To write a program to inter two strings & concatenate them

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

```
int main()
```

{

```
char str1[100] = "Breaks",  
      str2[100] = "World";  
  
char str3[100],  
     int i = 0, j = 0;
```

```
printf("The first string: %s\n", str1);  
printf("The second string: %s\n", str2);  
while (str1[i] != '\0' || str2[j] != '\0')  
    {
```

```
str3[i+j] = str1[i];
```

```
i++;
```

```
j++;
```

{

```
i = 0;
```

```
while (str1[i] != '\0')  
    {  
        str3[i] = str1[i];  
        i++;  
    }  
    str3[i] = '\0';
```

```
printf("%s", str3);  
printf("\n Concatenated string  
is: %s", str3);
```

o) Write a program to insert an element in Binary tree using array.

```
#include <conio.h>
#include <iostream.h>
int main()
{
    int j, n;
```

```
    int node[100];
    clrscr();
    cout << "Enter the number of elements : ";
    cin >> n;
    for (j = 1; j <= n; j++)
    {
        cout << "Enter the node : ";
        cin >> node[j];
    }
```

```
    cout << "Enter the node to be inserted : ";
    int d;
    cout << "Enter the node to be inserted : ";
    cout << endl;
```

```
    cout << "The members are : \n";
    for (j = 1; j <= n; j++)
    {
        cout << node[j] << " ";
    }
```

```
    cout << endl;
    getch();
```

Q. Write a program to implement of Binary Tree.

```
#include < stdio.h>
#include < conio.h>
int main()
{
    int i, n;
    int m[100];
    clrscr();
    printf("Enter the number of Element in Binary: ");
    scanf("%d", &n);
    for (i = 1; i <= n; i++)
    {
        printf("Enter the node: ");
        scanf("%d", &m[i]);
        if (m[i] == -1)
            break;
    }
    printf("Enter the node: ");
    scanf("%d", &n);
    for (i = 1; i <= n; i++)
    {
        printf("Enter the number of left child: ");
        scanf("%d", &m[i * 2]);
        printf("Enter the number of right child: ");
        scanf("%d", &m[i * 2 + 1]);
    }
}
```

general;

Date / /
Page No. 90

to write a program to delete an element from a binary tree

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int tree[32], choice,
        choice();
    printf("Enter the root node: ");
    scanf("%d", &tree[1]);
    printf("Left Child: ");
    scanf("%d", &tree[2]);
    printf("Right Child: ");
    scanf("%d", &tree[3]);
    printf("Left Child: ");
    scanf("%d", &tree[4]);
    printf("Right Child: ");
    scanf("%d", &tree[5]);
    printf("Delete node: ");
    scanf("%d", &choice);
    for (int i = 1; i < 3; i++)
    {
        if (choice == 1)
            tree[i] = 0;
        else if (choice == 2)
            tree[i] = 1;
        else if (choice == 3)
            tree[i] = 2;
    }
}
```

Output:
Enter the root node: 10
Left Child: 1 Right Child: 2
Left Child: 1 Right Child: 2
Delete node: 10
1
2
3
4
5
0
0
0
0
0

tree[1] = 0;

3

Date / /
Page No. 12

$$\log [2] = 0.3$$

१५३

$$\text{true}[\varphi] = \alpha.$$

privileges after delegation of M&S : (iii) ;
Year (i.e.) 1993-94

getcho (3),