

# DS PRACTICAL NO : 2

## [ ARRAY INSERTION AND DELETION ]

[A1] : Create an array of size n and write a program to insert an element at beginning.

**Program : -**

```
#include <stdio.h> int main() {    int arr[5] = {10, 20, 30, 40};

int n = 4;    int new    printf("Enter the element you want to
insert at beginning: ");    scanf("%d", &new)

    for (int i = n; i > 0; i--){
arr[i] = arr[i - 1];

    }

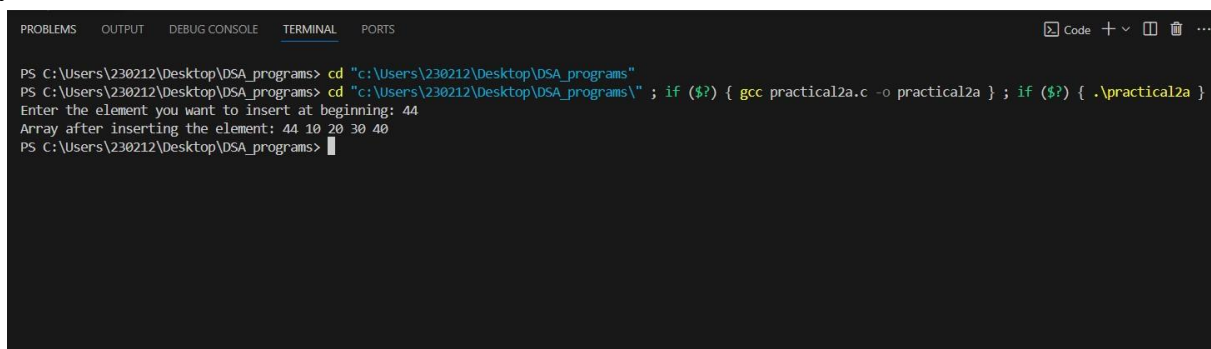
arr[0] = new;    n++;    printf("Array after
inserting the element: ");    for (int i = 0; i < n;
i++)

    {

        printf("%d ", arr[i]);

    } return 0;

}
```

A screenshot of a Windows terminal window with a dark background. The terminal shows the execution of a C program. The user navigates to the directory 'c:\Users\230212\Desktop\DSA\_programs' and runs 'gcc practical2a.c -o practical2a'. Then, they run './practical2a'. The program prompts 'Enter the element you want to insert at beginning:' and the user enters '44'. The program then outputs 'Array after inserting the element: 44 10 20 30 40'. The terminal window has tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL', and 'PORTS'. The 'TERMINAL' tab is active. There are icons for 'Code', a plus sign, a minus sign, a square, a trash can, and a three-dot menu in the top right corner of the terminal window.

```
PS C:\Users\230212\Desktop\DSA_programs> cd "c:\Users\230212\Desktop\DSA_programs"
PS C:\Users\230212\Desktop\DSA_programs> cd "c:\Users\230212\Desktop\DSA_programs\" ; if ($?) { gcc practical2a.c -o practical2a } ; if ($?) { .\practical2a }
Enter the element you want to insert at beginning: 44
Array after inserting the element: 44 10 20 30 40
PS C:\Users\230212\Desktop\DSA_programs> █
```

[A2] : Create an array of size n and write a program to insert an element at end of the program .

## DS PRACTICAL NO : 2

**Program :-** #include

<stdio.h>

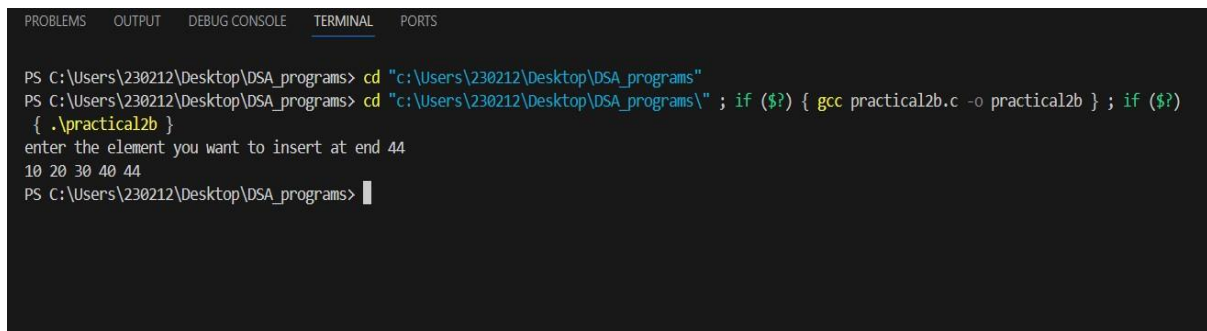
```
int main() {    int arr[5] = {10, 20, 30, 40};    int n = 4;

    int new;    printf("enter the element you want to insert
at end ");    scanf("%d",&new);    arr[n] = new;    n++;

    for (int i = 0; i < n; i++) {
printf("%d ", arr[i]);

    }

    return 0;
}
```



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\230212\Desktop\DSA_programs> cd "c:\Users\230212\Desktop\DSA_programs"
PS C:\Users\230212\Desktop\DSA_programs> cd "c:\Users\230212\Desktop\DSA_programs\" ; if ($?) { gcc practical2b.c -o practical2b } ; if ($?)
{ .\practical2b }
enter the element you want to insert at end 44
10 20 30 40 44
PS C:\Users\230212\Desktop\DSA_programs> |
```

[A3] : : Create an array of size n and write a program to insert an element at any position of the program

**Program : -** #include

<stdio.h>

```
int main() {    int array[10] =
{10,20,40,50};    int size = 4;

    int new_element = 30;

    int position = 2;
```

## DS PRACTICAL NO : 2

```
for (int i = size; i > position; i--)  
{  
    array[i] = array[i - 1];  
}  
array[position] = new_element;  
size++;
```

```
printf("Array after insertion: ");
```

```
for (int i = 0; i < size; i++)  
{  
    printf("%d ", array[i]);  
}  
printf("\n");  
  
return 0;  
}
```



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
PS C:\Users\230212\Desktop\DSA_programs> cd "c:\Users\230212\Desktop\DSA_programs"  
PS C:\Users\230212\Desktop\DSA_programs> cd "c:\Users\230212\Desktop\DSA_programs\" ; if ($?) { gcc practical2c.c -o practical2c } ; if ($?) { .\practical2c }  
Array after insertion: 10 20 30 40 50  
PS C:\Users\230212\Desktop\DSA_programs>
```

[B1] : Create an array of size n and write a program to delete an element at beginning.

**Program :-** #include

<stdio.h> int main()

## DS PRACTICAL NO : 2

```
{  
    int array[100] = {10, 20, 30, 40, 50};  
    int size = 5;    for (int i = 0; i < size - 1;  
    i++)  
    {  
        array[i] = array[i + 1];  
    }  
    size--;    printf("Array after  
deletion: ");    for (int i = 0; i <  
size; i++)  
    {  
        printf("%d ", array[i]);  
    }  
    printf("\n");  
  
    return 0;  
}
```



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
PS C:\Users\230212\Desktop\DSA_programs> cd "c:\Users\230212\Desktop\DSA_programs"  
PS C:\Users\230212\Desktop\DSA_programs> cd "c:\Users\230212\Desktop\DSA_programs\" ; if ($?) { gcc practical2D.c -o practical2D } ; if ($?)  
{ .\practical2D }  
Array after deletion: 20 30 40 50  
PS C:\Users\230212\Desktop\DSA_programs>
```

[B2] : Create an array of size n and write a program to delete an element at end of the program .

**Program:-** #include

<stdio.h> int main()

## DS PRACTICAL NO : 2

```
{  
    int array[100] = {10, 20, 30, 40, 50};  
    int size = 5;    size--;    printf("Array  
after deletion: ");    for (int i = 0; i <  
size; i++)  
    {  
        printf("%d ", array[i]);  
    }  
    printf("\n");  
    return 0;  
}
```



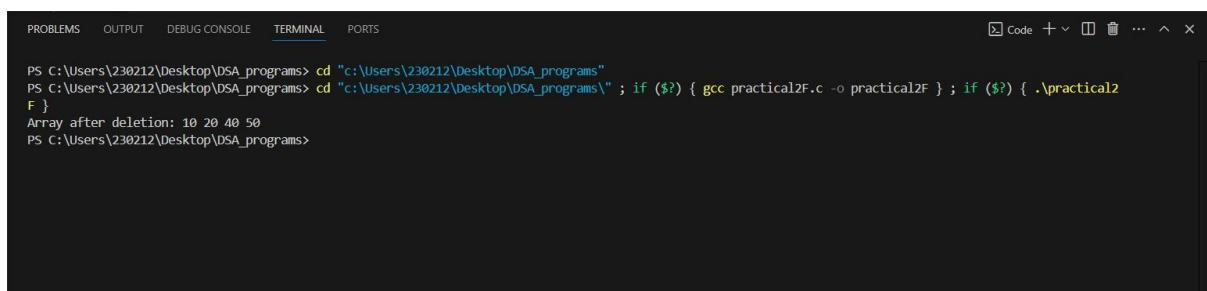
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
PS C:\Users\230212\Desktop\DSA_programs> cd "c:\Users\230212\Desktop\DSA_programs"  
PS C:\Users\230212\Desktop\DSA_programs> cd "c:\Users\230212\Desktop\DSA_programs\" ; if ($?) { gcc practical2E.c -o practical2E } ; if  
E }  
Array after deletion: 10 20 30 40  
PS C:\Users\230212\Desktop\DSA_programs>
```

[B3] : Create an array of size n and write a program to delete an element at any position of the program **Program :-**

```
#include <stdio.h> int main() {    int  
array[100] = {10, 20, 30, 40, 50};    int  
size = 5;    int position = 2;  
  
    for (int i = position; i < size - 1; i++)  
    {  
        array[i] = array[i + 1];  
    }  
    size--;
```

## DS PRACTICAL NO : 2

```
printf("Array after deletion: ");  
for (int i = 0; i < size; i++)  
{  
    printf("%d ", array[i]);  
}  
printf("\n");  
return 0;  
}
```



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
PS C:\Users\230212\Desktop\DSA_programs> cd "c:\Users\230212\Desktop\DSA_programs"  
PS C:\Users\230212\Desktop\DSA_programs> cd "c:\Users\230212\Desktop\DSA_programs\" ; if ($?) { gcc practical2F.c -o practical2F } ; if ($?) { .\practical2F }  
Array after deletion: 10 20 40 50  
PS C:\Users\230212\Desktop\DSA_programs>
```

GITHUB LINK OF PRACTICAL NO 2 :

<https://github.com/AmolNagargoje04/Data-Structure-practical>