

Name: Vedant Mhatre  
Moodle ID: 18102055  
Roll Number: 31

Modification 1:

```
#include <stdio.h>

int main()
{
    int array[100], n, c, d, position, swap;

    printf("Enter number of elements\n");
    scanf("%d", &n);

    printf("Enter %d integers\n", n);

    for (c = 0; c < n; c++)
        scanf("%d", &array[c]);

    for (c = n-1; c > 0; c--)
    {
        position = c;

        for (d = c - 1; d > -1; d--)
        {
            if (array[position] > array[d])
                position = d;
        }
        if (position != c)
        {
            swap = array[c];
            array[c] = array[position];
            array[position] = swap;
        }
    }

    printf("Sorted list in descending order:\n");

    for (c = 0; c < n; c++)
        printf("%d\n", array[c]);

    return 0;
}
```

Output:

```
vedant@hp:~/Documents/College/A0A/Exp2$ gcc desc.c -o desc
vedant@hp:~/Documents/College/A0A/Exp2$ ./desc
Enter number of elements
5
Enter 5 integers
2
55
33
76
22
Sorted list in descending order:
76
55
33
22
2
vedant@hp:~/Documents/College/A0A/Exp2$
```

Modification 2:

Code:

```
#include <stdio.h>

void sort(int array[], int n)
{
    int c,d,position,swap;
    for (c = n-1; c > 0; c--)
    {
        position = c;

        for (d = c - 1; d > -1; d--)
        {
            if (array[position] > array[d])
                position = d;
        }
        if (position != c)
        {
            swap = array[c];
            array[c] = array[position];
            array[position] = swap;
        }
    }
}
```

```
}  
}  
}
```

```
int main()  
{  
    int array[100], n, c;  
  
    printf("Enter number of elements\n");  
    scanf("%d", &n);  
  
    printf("Enter %d integers\n", n);  
  
    for (c = 0; c < n; c++)  
        scanf("%d", &array[c]);  
  
    sort(array,n);  
  
    printf("Sorted list in descending order:\n");  
  
    for (c = 0; c < n; c++)  
        printf("%d\n", array[c]);  
  
    return 0;  
}
```

Output:

```
vedant@hp:~/Documents/College/AOA/Exp2$ gcc mod2.c -o mod2  
vedant@hp:~/Documents/College/AOA/Exp2$ ./mod2  
Enter number of elements  
4  
Enter 4 integers  
12  
32  
11  
45  
Sorted list in descending order:  
45  
32  
12  
11  
vedant@hp:~/Documents/College/AOA/Exp2$
```

Modification 3:

Code:

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

```
#include<stdlib.h>
```

```
struct node{
    int data;
    int id;
    struct node *next;
};
```

```
struct node *top = NULL;
```

```
struct node *ptr = NULL;
```

```
void insert()
```

```
{
    struct node *new_node = (struct node *) malloc(sizeof(struct node));
    scanf("%d",&new_node->data);
    if (top == NULL)
        new_node->id = 0;
    else
        new_node->id = (top->id) + 1;
    new_node->next = top;
    top = new_node;
}
```

```
void print()
```

```
{
    ptr = top;
    while(ptr!=NULL)
    {
        printf("%d\n",ptr->data);
        ptr = ptr->next;
    }
}
```

```
void ptrAtId(int pid)
```

```
{
    ptr = top;
    while(ptr->id!=pid)
        ptr = ptr->next;
}
```

```
int ptrData(int pid)
```

```
{  
    ptrAtId(pid);  
    pid = ptr->data;  
    return pid;  
}
```

```
void changeData(int pid,int value)
```

```
{  
    ptrAtId(pid);  
    ptr->data = value;  
}
```

```
void sort(struct node *top, int *n)
```

```
{  
  
    int *c = (int *)malloc(sizeof(int));  
    int *d = (int *)malloc(sizeof(int));  
    int *position = (int *)malloc(sizeof(int));  
    int *swap = (int *)malloc(sizeof(int));  
  
    for (*c = *n-1; *c > 0; *c=*c-1)  
    {  
        *position = *c;  
  
        for (*d = *c - 1; *d > -1; *d=*d-1)  
        {  
            if ( ptrData(*position) > ptrData(*d))  
                *position = *d;  
        }  
        if (*position != *c)  
        {  
            *swap = ptrData(*c);  
            changeData(*c,ptrData(*position));  
            changeData(*position,*swap);  
        }  
    }  
}
```

```
void main()
{
    int *n = (int *)malloc(sizeof(int));
    printf("Enter number of elements\n");
    scanf("%d", n);
    printf("Enter %d values\n", *n);

    int *c = (int *)malloc(sizeof(int));
    for(*c=0;*c<*n;*c=*c+1)
        insert();
    printf("Before Sorting:\n");
    print();
    sort(top,n);
    printf("After Sorting:\n");
    print();
}
```

Output:

```
vedant@hp:~/Documents/College/A0A/Exp2$ gcc mod3.c -o mod3
vedant@hp:~/Documents/College/A0A/Exp2$ ./mod3
Enter number of elements
5
Enter 5 values
11
34
21
45
12
Before Sorting:
12
45
21
34
11
After Sorting:
11
12
21
34
45
vedant@hp:~/Documents/College/A0A/Exp2$
```