1) Program to swap two numbers using pointers

```
#include<stdio.h>
void swap(int *a,int *b);
void main()
{
  int a=10;
  int b=45;
  printf("the values before swapping=%d %d\n",a,b);
  swap(&a,&b);
  printf(" the values after swapping=%d %d\n",a,b);
}
void swap(int *a,int *b)
{
    int temp=*a;
    temp=*a;
    *a=*b;
    *b=temp;
}
```

```
the values before swapping=10 45
 the values after swapping=45 10
```

2)Program to show dynamic memory allocation

```
#include<stdio.h>
#include<stdlib.h>
void main()
{
    int *p,*q;
    int n;
    int i;
    printf("read n\n");
```

```
    scanf("%d",&n);
    p=(int*)malloc(n*sizeof(int));
    printf("enter %d elements\n",n);
    for(i=0;i<n;i++)
    {
        scanf("%d",p+i);
    }
    q=(int*)calloc(n,sizeof(int));
    printf("enter %d elements\n",n);
    for(i=0;i<n;i++)
    {
        scanf("%d",q+i);
    }
    p=realloc(p,5*sizeof(int));
    printf("enter five elements\n");
    for(i=0;i<5;i++)
    {
        scanf("%d",p+i);
    }
    free(p);
    free(q);
}
```

```
read n
3
enter 3 elements
10
20
30
enter 3 elements
12
14
15
enter five elements
8
2
4
6
9
```

3)Stack implementation

```
#include <stdio.h>
#define max 5
int top = -1;
int s[max];
void push(int value)
{
  if (top == max - 1)
  {
     printf("stack overflow ");

  }
  else
  {
   top = top + 1;
   s[top] = value;
   }
}
void pop()
```

```
{
  int value;
  if (top == -1)
  {
     printf("stack is underflow \n");
  }
  else
  {
    value = s[top];
    top = top - 1;
    printf("\n%d is popped\n", value);
  }

 }
void isempty()
{
  if (top == -1)
{
   printf("stack is empty\n");
}
}
void isfull()
{
if (top == max - 1)
printf("stack is full\n");
}
void display()
{
  if (top == -1)
     printf("stack is underflow\n");
  else
  {
    printf("\n stack elements are:");
    for (int i = 0; i <= top; i++)
    printf("%d\t", s[i]); }
```

```
}
void main()
{
   int value;
   int no;
   printf("enter a no:");
   scanf("%d", &no); push(no);
   printf("enter a no:");
   scanf("%d", &no);
   push(no);
   printf("enter a no:");
   scanf("%d", &no);
   push(no);
   printf("enter a no:");
   scanf("%d", &no);
   push(no);
   printf("enter a no:");
   scanf("%d", &no);
   push(no);
   printf("enter a no:");
   scanf("%d", &no);
   push(no);
   display();
   pop();
   pop();
   pop();
   pop();
   isempty();
   isfull();
   display();
   pop();
   pop();
}
```

```
enter a no:20
enter a no:14
enter a no:12
enter a no:87
enter a no:34
enter a no:12
stack overflow cant push
 stack elements are:20  14      12      87      34
34 is popped

87 is popped

12 is popped

14 is popped

 stack elements are:20
20 is popped
stack is underflow cant pop
```