2) Handling strings

Sol:

```
#include <stdio.h>
int main()
{
   char str[10];
   printf("Enter the string\n");
   scanf("%s",str);
   int i=0,lowr=0,uppr=0,spl=0,num=0;
   int len=0;
   while(str[i]!='\0')
   {
      if(str[i]>='a' && str[i]<='z' )
      {
         lowr++;
      }
      if(str[i]>='A' && str[i]<='Z')
      {
         uppr++;
      }
      if(str[i]>=48 && str[i]<=57)
      {
         num++;
      }
      if((str[i]>=33 && str[i]<=47) || (str[i]>=58 && str[i]<64) || (str[i]>=91 && str[i]<=96) ||
(str[i]>=123 && str[i]<=126))
      {
         spl++;
      }
      len++;
      i++;
   }
   if(len<8)
      printf("Must be of length 8");
   else if(lowr==0)
      printf("Must contain atleast one lower-case");
   else if(uppr==0)
      printf("Must contain atleast one upper-case");
   else if(spl==0)
      printf("Must contain atleast one special-case");
   else
      printf("You entered a valid string!!");
   return 0;
}
```

3)Array traversal using pointers

Sol:

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

```
int main()
{
    int arr[5]={1,2,3,4,5};

    for(int *ptr=&arr[0];ptr<=&arr[4];ptr++)
    {
        printf("%d ",*ptr);

    }
    return 0;
}
```

4) string permutation using pointers:

Sol:

```
#include<stdio.h>
void permute(char *str,int left,int right);
int main(){
    char str[10];
    printf("enter the string\n");
    scanf("%[^\n]s",str);
    int i;
    int len =0,num=0,lowr=0,uppr=0,spl=0;
    for(i=0;str[i]!=0;i++){
        len++;
    }
    if(len!=8){
        printf("enter character length of 8 \n");
    }
    else{
        for(i=0;i<len;i++){
            if(str[i]>='0' && str[i]<='9'){
                num++;
            }
            else if(str[i]>='a' &&str[i]<='z'){
                lowr++;

            }
            else if(str[i]>='A' && str[i]<='Z'){
                uppr++;
            }
            else{
                spl++;
            }
        }
        if(num!=0 && lowr!=0 && uppr!=0 && spl!=0){
            printf("given testcases are satisfied\n");
            printf("the possible permutations are ;\n");
            permute(str,0,len-1);
        }
        else{
            printf("given testcases are not satisfied\n");
        }

        return 0;
    }


}
void permute(char *str,int left,int right){
    int i,j;
    char temp;
    for(i=left;i<=right;i++){
        for(j=i+1;j<right;j++){
            temp=str[i];
            str[i]=str[j];
            str[j]=temp;
            permute(str,i+1,right);
            temp=str[i];
            str[i]=str[j];
            str[j]=temp;

        }
    }
    printf("%s\n",str);
}
```

5)Dynamic creation of array of integers

Sol:

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
    int size ;
    printf("Enter size of array \n");
    scanf("%d",&size);
    int *arr;
    arr=(int*)malloc(size*sizeof(int));
    int i;
    printf("Enter %d elements into the array \n",size);
    for(i=0;i<size;i++)
    {
        scanf("%d",&arr[i]);
    }
    int sum =0;
    for(i=0;i<size;i++)
    {
        sum +=arr[i];
    }
    printf("sum of array elements: %d\n ",sum);
    free(arr);
    return 0;
}
```