1. Calculate the area of square or circle based on the shape ‘S’ for Square and ‘C’ for Circle.

Code

```
#include<stdio.h>
int main()
{
        char Shape;
        int size;
        float area;
        const float pi = 3.14;
        printf("Enter 'S' for square 'C' for circle:");
        scanf("%c",&Shape);
        printf("Enter the size:");
        scanf("%d",&size);
        switch(Shape)
        {
                case 'S':
                        area = size*size;
                        printf("Area of Square: %.0f",area);
                        break;
                case 'C':
                        area = pi*size*size;
                        printf("Area of Circle : %.2f",area);
                        break;
                default:
                        printf("Invalid input");
                        break;
        }
        return 0;
```

}

```
Enter 'S' for square 'C' for circle:C
Enter the size:4
Area of Circle : 50.24
--------------------------------
Process exited after 4.456 seconds with return value 0
Press any key to continue . . .
```

2. Given a sorted array having duplicate elements. Print the elements with its frequency having more than one appearance.

Code

```
#include<stdio.h>
void findDuplicate(int n,int arr[])
{
        int element=0,i,j,count;
        for(i=0;i<n;i++)
        {
          count = 0;
           if(element!=arr[i])
           {
                element = arr[i];
                for(j=i;j<n;j++)
                {
                    if(arr[j]==element)
                   {
                    count++;
                    }
                }
                if(count>1)
                {
                        printf("%d->%d   ",element,count);
                }
          }
      }
```

```
}
int main()
{
	int n,i;
	printf("Enter the value of n:");
	scanf("%d",&n);
	int arr[n];
	printf("Enter the %d array elements:\n",n);
	for(i=0;i<n;i++)
	{
		scanf("%d",&arr[i]);
	}
	findDuplicate(n,arr);
	return 0;
}
```

Output

```
Enter the value of n:12
Enter the 12 array elements:
1
1
1
2
4
4
4
5
6
9
9
1->3   4->4   9->2
--------------------------------
Process exited after 17.35 seconds with return value 0
Press any key to continue . . .
```

3 . 3. Given a sentence and screen length. Justify the sentence according to the screen length by replacing space with stars

Code

```
#include <stdio.h>
void justifySentence(char sentence[], int screenLength) {
   int length = 0;
   int i,j;
   int spaceCount = 0;
   for ( i = 0; sentence[i] != '\0'; i++) {
      if (sentence[i] == ' ') {
         spaceCount++;
      }
      length++;
   }
   int extraSpaces = screenLength - length;
   int spacesBetweenWords = spaceCount > 0 ? extraSpaces / spaceCount : 0;
   int remainingSpaces = spaceCount > 0 ? extraSpaces % spaceCount : 0;
   for (i = 0; sentence[i] != '\0'; i++) {
      if (sentence[i] == ' ') {
         for (j = 0; j < spacesBetweenWords+1; j++) {
            printf("*");
         }
      if (remainingSpaces > 0) {
            printf("*");
            remainingSpaces--;
         }
      } else {
         printf("%c", sentence[i]);
      }
   }
printf("\n");
}
```

```
int main() {
    char sentence[30];
    int screenLength;
    printf("Enter the sentence: ");
    gets(sentence);
    printf("Enter the screen length: ");
    scanf("%d", &screenLength);
    justifySentence(sentence, screenLength);
    return 0;
}
```

Output