Q1) Write a function to calculate length of the string

```
#include<stdio.h>

int fun(char str[]);

int main()
{
   char str[150];

   printf("Enter string = ");
   fgets(str,150,stdin);

   int len = fun(str);

   printf("\nlength of the string = %d", len);

}

int fun(char str[])
{
   int i=0;

   while((str[i] != 10) && (str[i] != '\0'))
```

```
        i++;


    return i;
}
```

Q2) Write a function to reverse a string.

```
#include<stdio.h>

char* fun(char []);

char rev[150];

int main(int argc, char *argv[])
{
        char str[150];

        printf("enter string :- ");
        fgets(str,150,stdin);

        char* rev = fun(str);

        printf("%s\t",rev);

}

char* fun(char str[])
{
   int i = 0;
```

```
    for(;(str[i]!=10) && (str[i]!='\0');i++);

    for(int j = 0; i>=0;i--,j++)
    {
        rev[j]=str[i];
    }

    return(rev);
}
```

Q3) Write a function to compare two strings.

```
#include<stdio.h>

void fun(char [], char []);

int main(int argc, char *argv[])
{
        char str1[150];
   char str2[150];

        printf("enter string 1 :- ");
        fgets(str1,150,stdin);

        printf("enter string 2 :- ");
        fgets(str2,150,stdin);

   fun(str1, str2);

}

void fun(char str1[], char str2[])
{
   int chk = 0;
```

```
    for(int i = 0; (str1[i] != '\0') || (str2[i] != '\0'); i++)
    {
        if(str1[i] != str2[i])
        {
            chk = 1;
            break;
        }
    }

    if(chk == 0)
    {
        printf("String is equal");
    }
    else
    {
        printf("String is not equal");
    }
}
```

Q4) Write a function to transform string into uppercase

```
#include<stdio.h>

void fun(char str[]);

int main()
{
   char str[200];
   int count = 0;

   printf("Enter String = ");
   fgets(str,200,stdin);

   fun(str);
}

void fun(char str[])
{
   for(int i=0; (str[i] != 10) && (str[i] != '\0'); i++)
      if((str[i] >= 97) && (str[i] <= 122))
         str[i] = str[i] - 32 ;

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

```
}
```

Q5) Write a function to transform a string into lowercase

```
#include<stdio.h>

void fun(char str[]);

int main()
{
   char str[200];
   int count = 0;

   printf("Enter String = ");
   fgets(str,200,stdin);

   fun(str);
}

void fun(char str[])
{
   for(int i=0; (str[i] != 10) && (str[i] != '\0'); i++)
      if((str[i] >= 65) && (str[i] <= 90))
         str[i] = str[i] + 32 ;

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

```
}
```

Q6) Write a function to check whether a given string is an alphanumeric string or not.

(Alphanumeric string must contain at least one alphabet and one digit)

```
#include<stdio.h>

void fun(char str[]);

int main()
{
    char str[200];
    int count = 0;

    printf("Enter String = ");
    fgets(str,200,stdin);

    fun(str);
}

void fun(char str[])
{
    int flag = -1;
```

```
for(int i=0; (str[i] != 10) && (str[i] != '\0'); i++)
{
   if((str[i] >= 65) && (str[i] <= 90) || (str[i] == 32));
   else if((str[i] >= 97) && (str[i] <= 122));
   else if((str[i] >= 48) && (str[i] <= 57))
   {
      flag = 1;
   }
   else
   {
      printf("This string is not alphanumeric");
      flag = 0;
      break;
   }
}

if(flag == 1)
{
   printf("This string is alphanumeric");
}
else if(flag == -1)
{
   printf("This string is not alphanumeric");
}
```

}

Q7) Write a function to check whether a given string is palindrome or not.

```
#include<stdio.h>

int fun(char str[]);

int main()
{
    char str[200];
    int count = 0;

    printf("Enter String = ");
    fgets(str,200,stdin);

    if(fun(str))
    {
        printf("\nstring is palindrome");
    }
    else
    {
        printf("\nstring is not palindrome");
    }
}
```

```
int fun(char str[])
{
   int i = 0;
   char rev[200];

   for(; (str[i]!=10) && (str[i]!='\0'); i++);

   i=i-1;

   for(int j = 0; i>=0; i--,j++)
   {
      rev[j]=str[i];
      printf("%d %c\n",j,rev[j]);
   }

   i = 0;

   for(; (str[i]!=10) && (str[i]!='\0'); i++)
   {
      if(str[i] == rev[i])
      {
         continue;
      }
```

```
        else
            return 0;
    }

    return 1;

}
```

Q8) Write a function to count words in a given string

```
#include<stdio.h>

void fun(char str[]);

int main()
{
   char str[200];
   int count = 0;

   printf("Enter String = ");
   fgets(str,200,stdin);

   fun(str);
}

void fun(char str[])
{
   int sp = 0, i = 0;

   if(str[0] == 32)
   {
      i++;
```

```
    }

    for(; str[i]!='\0'; i++)
    {
        if((str[i-1] != 32) && (str[i] == 32 || str[i] == 10))
        {
            sp = sp + 1;
        }
    }

    printf("Total Words = %d", sp);
}
```

Q9) Write a function to reverse a string word wise. (For example if the given string is

"Mysirg Education Services" then the resulting string should be "Services Education Mysirg")

```
#include<stdio.h>

void fun(char str[]);

int main()
{
    char str[200];
    int count = 0;

    printf("Enter String = ");
    fgets(str,200,stdin);

    fun(str);
}

void fun(char str[])
{
    int length = 0;
    char rev[200];
```

```
int start_ptr, traverse_ptr, space_ptr, rev_ptr = 199;

rev[rev_ptr] = '\0';


//find length of string
while(str[length] != '\0')
{
   length++;
}

start_ptr = traverse_ptr = 0 ; //initialize pointer for first word

while(str[traverse_ptr] != '\0')
{
   if(str[traverse_ptr] == 32 || str[traverse_ptr] == 10)
   {
      space_ptr = traverse_ptr; // space_pointer point the space

      while(traverse_ptr != start_ptr)
      {
         rev_ptr = rev_ptr - 1;
         rev[rev_ptr] = str[traverse_ptr - 1];
```

```
            traverse_ptr = traverse_ptr - 1;
        }
        rev_ptr = rev_ptr - 1;
        rev[rev_ptr] = str[space_ptr];
        start_ptr = traverse_ptr = space_ptr + 1;
      }
      else
      {
        traverse_ptr = traverse_ptr + 1;
      }
    }

    while(rev[rev_ptr] != '\0')
    {
      printf("%c", rev[rev_ptr]);
      rev_ptr = rev_ptr + 1;
    }
}
```

Q10) Write a function to find the repeated character in a given string.

```
#include<stdio.h>

void fun(char []);

int main(int argc, char *argv[])
{
        char str[150];

        printf("enter string :- ");
        fgets(str,150,stdin);

        fun(str);
}

void fun(char str[])
{
   char tmp[200];
   int trv, flag = 0;

   for(int i = 0; str[i] != '\0'; i++)
   {
```

```
trv = i + 1;
if(str[i] == tmp[i])
{
}
else
{
   while(str[trv] != '\0')
   {
      if(str[i] == str[trv])
      {
         tmp[trv] = str[trv];
         trv++;
         flag = 1;
      }
      else
      {
         trv++;
      }
   }
   if(flag == 1)
   {
      printf("\n%c ", str[i]);
   }
   flag = 0;
```

```
        }
    }
}
```