Assignment-18

①
```
#include<stdio.h>
void length(char a[]);
int main()
{
    char str[50];
    printf("Enter a string");
    fgets(str, 50, stdin);
    length(str);
    return 0;
}
void length(char a[])
{
        printf("length of string = %d", strlen(a)-1),
}
```

②
```
# include<stdio.h>
void reverse(char a[]);
int main()
{
        char str[50];
        printf("Enter a string");
        fgets(str, 50, stdin);
        reverse(str);
        return0;
  }
  void reverse(char a[])
  {
  } printf("reverse of string= %s", strrev(a));
```

Assignment-18

(1)

```
#include<stdio.h>
void length(char a[]);
int main()
{
     char str[50];
     printf("Enter a string");
     fgets(str,50,stdin);
     length(str);
     return 0;
}
void length(char a[])
{
        printf("length of string=%d",strlen(a)-1),
}
```

(2)

```
#include<stdio.h>
void reverse(char a[]);
int main()
{
       char str[50];
       printf("Enter a string");
       fgets(str,50,stdin);
       reverse(str);
       return0;
}
void reverse(char a[])
{
 printf("reverse of string=%s",strrev(a));
}
```

```
③ #include <stdio.h>
 void compare(char a[], char t[]);
 int main()
 {
     char str[50];
     printf("Enter a string");
     fgets(str, 50, stdin);
     char p[50];
     printf("enter a string");
     fgets(p, 50, stdin);
     compare(str, p);
     return 0;
  }
  void compare(char a[], char t[])
  {
       int m;
       m = strcmp(a, t);
       If (m==0)
            printf("same string");
       elseif(m==1)
            printf("opposite direction order");
       else
            printf("dictionary order");
   }
```

```
④ #include <stdio.h>
~~intmain~~
void uppercase (char a[]);
int main()
{
      char str[50];
      printf("enter a string");
      fgets(str, 50, stdin);
      uppercase(str);
      return0;
}
void uppercase(char a[])
{   printf("%s",strupr(a));
}
```

```
⑤ #include <stdio.h>
void lowercase (char a[]);
intmain()
{   char str[50];
    printf("Enter a string");
    fgets (str, 50, stdin);
    lowercase(~~@~~str);
    return0;
}
void lowercase(char a[])
{       printf("%s", strlwr(a));
}
```

```
6 #include <stdio.h>
  void alphanumeric (char a[])
  int main()
  {
     char str[50];
     printf("Enter a string");
     fgets(str, 50, stdin);
     alphanumeric(str);
     return 0;
  }
  void alphanumeric(char a[])
  {
       int i,j
       for(i=0; a[i], i++)
       {   If(a[i]==1||a[i]==2||a[i]==3||a[i]
           ==4||a[i]==5||a[i]==6||a[i]==7||
             a[i]==8|a[i]==9)
             {
             }
             printf("Alphanumeric");
       }
       printf("Not alphanumeri");

  }
```

```
(7) # include <stdio.h>
void palindrom (char a[]);
int main()
{
      char str[30];
      printf("enter a string");
      fgets(str, 30, stdin);
       palindrom(str);
       return 0;
}
void palindrom (char a[])
{
        int l, i;
        l = strlen(a);
        for (i=0; i< l/2 ; i++)
         {
               If (a[i] != a[l-1-i])
                          printf("Not palindrom")
                   break;
         }
        If (i==l/2)
                  printf("palindrom");

}
```

```
⑧ #include<stdio.h>
void cauntword(char b[]);
int main()
{
        char str[50];
        printf("enter a string");
        fgets(str, 50, stdin);
        cauntword(str);
        return 0;
  }
  void cauntword(char b[])
  {
          int l, i, caunt=0;
          l=strlen(b);
          for(i=0; i<=l; i++)
          {       If(b[i]==' ')
                          caunt++;
            }
          printf("%d", caunt);
    }
```

```
(10) #include <stdio.h>
 vaid repeat(char p[]);
 int main()
 {
       char str[200];
       printf("enter a ~~array~~ string")
       fgets(str, 200, stdin);
       repeat(arr);
       return0;
  }
  vaid repeat(char p[])
  {
        int i, J, l
        l = strlen(p);
        for(i=0; i<l-1; i++)
        {    for(J=1; J<l ; J++)
             {
                  If (str[i] == str[J])
                       printf("c", str[i]);
             }
        }
  }
```