Practical No:3

Program

```
#include<stdio.h>
#include<string.h>
#define MAX 200
void longestWord(char s[]){
   char w[50],l[50];int i=0,j=0,m=0;
   while(1){
      if(s[i]!=' '&&s[i]!='\0')w[j++]=s[i];
      else{w[j]='\0';if(j>m){m=j;strcpy(l,w);}j=0;}
      if(s[i]=='\0')break;i++;
   }
   printf("\nLongest word: %s (Length=%d)\n",l,m);
}
void charFrequency(char s[],char c){
   int cnt=0;for(int i=0;s[i];i++)if(s[i]==c)cnt++;
   printf("\nFrequency of '%c'=%d\n",c,cnt);
}
void checkPalindrome(char s[]){
   int n=strlen(s),f=1;
   for(int i=0,j=n-1;i<j;i++,j--)if(s[i]!=s[j]){f=0;break;}
   printf(f?"\nPalindrome\n":"\nNot Palindrome\n");
}
void substringIndex(char s[],char sub[]){
   char *p=strstr(s,sub);
   printf(p?"\nSubstring at index %ld\n": "\nNot found\n",p-s);
}
void wordOccurrences(char s[]){
   char t[MAX];strcpy(t,s);char *w[50];int c[50]={0},n=0;
   char *tok=strtok(t," ");
   while(tok){int f=0;for(int i=0;i<n;i++)if(!strcmp(w[i],tok)){c[i]++;f=1;break;}
```

```
        if(!f){w[n]=strdup(tok);c[n]=1;n++;}tok=strtok(NULL," ");
    }
    printf("\nWord occurrences:\n");for(int i=0;i<n;i++)printf("%s:%d\n",w[i],c[i]);
}
int main(){
    char s[MAX],sub[50],c;
    printf("Enter string: ");fgets(s,MAX,stdin);s[strcspn(s,"\n")]=0;
    longestWord(s);
    printf("Enter char: ");scanf("%c",&c);getchar();charFrequency(s,c);
    checkPalindrome(s);
    printf("Enter substring: ");fgets(sub,50,stdin);sub[strcspn(sub,"\n")]=0;substringIndex(s,sub);
    wordOccurrences(s);
}
```

Output:

```
Enter a string: mayur

Longest word: mayur (Length = 5)
Enter character to find frequency: a

Frequency of 'a' = 1

String is NOT a Palindrome
Enter substring to find index: yu

Substring first appears at index 2

Word occurrences:
mayur : 1


=== Code Execution Successful ===
```