X

![swayam logo](https://swayam.gov.in) **(https://swayam.gov.in)**   **(https://swayam.gov.in/nc_details/NPTEL)**

200801168@rajalakshmi.edu.in ⌄

NPTEL (https://swayam.gov.in/explorer?ncCode=NPTEL) » Problem Solving Through Programming In C (course)

≡

Week 7 : Programming Assignment 4

Due on 2023-09-14, 23:59 IST

Write a C program to print Largest and Smallest Word from a given sentence. If there are two or more words of same length, then the first one is considered. A single letter in the sentence is also consider as a word.

Private Test cases used for evaluation	Input	Expected Output	Actual Output	Status
Test Case 1	AICTE Approved FDP Course.	Largest Word is: Approved\n Smallest word is: FDP	Largest Word is: Approved\n Smallest word is: FDP\n	Passed

The due date for submitting this assignment has passed.
1 out of 1 tests passed.
You scored 100.0/100.

Assignment submitted on 2023-09-13, 20:21 IST
Your last recorded submission was :

```
1  #include<stdio.h>
2  #include<string.h>
3  int main()
4  {
5      char str[100]={0},substr[100][100]={0};
6  //str[100] is for storing the sentence and substr[50][50] is for storing each
7
8  scanf("%[^\n]s", str); //Accepts the sentence from the test case data.
9
10 /* Complete the program to get the desired output.
11 The print statement should be as below
12
13 printf("Largest Word is: %s\nSmallest word is: %s\n", -------,--------);
14
15 */
```

Course outline

How does an NPTEL online course work? ()

Week 0 : ()

Week 1 ()

Week 2 ()

Week 3 ()

Week 4 ()

Week 5 ()

```c
16  int i=0,j=0,k=0,a,minIndex=0,maxIndex=0,max=0,min=0;
17  char c;
18  while(str[k]!='\0')  //for splitting sentence into words
19      {
20          j=0;
21          while(str[k]!=' '&&str[k]!='\0' && str[k]!='.')
22          {
23              substr[i][j]=str[k];
24              k++;
25              j++;
26          }
27          substr[i][j]='\0';
28          i++;
29          if(str[k]!='\0')
30          {
31              k++;
32          }
33      }
34      int len=i;
35      max=strlen(substr[0]);
36      min=strlen(substr[0]);
37
38      //After splitting getting length of string and finding its index having m
39      for(i=0;i<len;i++)
40      {
41          a=strlen(substr[i]);
42          if(a>max)
43          {
44              max=a;
45              maxIndex=i;
46          }
47          if(a<min)
48          {
49              min=a;
50              minIndex=i;
51          }
52      }
53      printf("Largest Word is: %s\nSmallest word is: %s\n",substr[maxIndex],subst
54      return 0;
55  }
56
```

◄ ▬▬▬▬▬▬▬▬▬▬▬   ►

Sample solutions (Provided by instructor)

```c
1   #include<stdio.h>
2   #include<string.h>
3   int main()
4   {
5       char str[100]={0},substr[100][100]={0};
6   //str[100] is for storing the sentence and substr[50][50] is for storing each
7
8   scanf("%[^\n]s", str); //Accepts the sentence from the test case data.
9
10  /* Complete the program to get the desired output.
11  The print statement should be as below
12
13  printf("Largest Word is: %s\nSmallest word is: %s\n", -------,--------);
14
15  */
16  int i=0,j=0,k=0,a,minIndex=0,maxIndex=0,max=0,min=0;
17  char c;
18  while(str[k]!='\0')  //for splitting sentence into words
19      {
20          j=0;
21          while(str[k]!=' '&&str[k]!='\0' && str[k]!='.')
22          {
23              substr[i][j]=str[k];
24              k++;
25              j++;
26          }
27          substr[i][j]='\0';
28          i++;
29          if(str[k]!='\0')
30          {
31              k++;
32          }
```

```
33        }
34        int len=i;
35        max=strlen(substr[0]);
36        min=strlen(substr[0]);
37
38        //After splitting getting length of string and finding its index having m
39        for(i=0;i<len;i++)
40        {
41            a=strlen(substr[i]);
42            if(a>max)
43              {
44                  max=a;
45                  maxIndex=i;
46              }
47            if(a<min)
48              {
49                  min=a;
50                  minIndex=i;
51              }
52        }
53      printf("Largest Word is: %s\nSmallest word is: %s\n",substr[maxIndex],subst
54      return 0;
55 }
```