APURV TODKAR
16BCB0048


# LAB EXPERIMENT 2

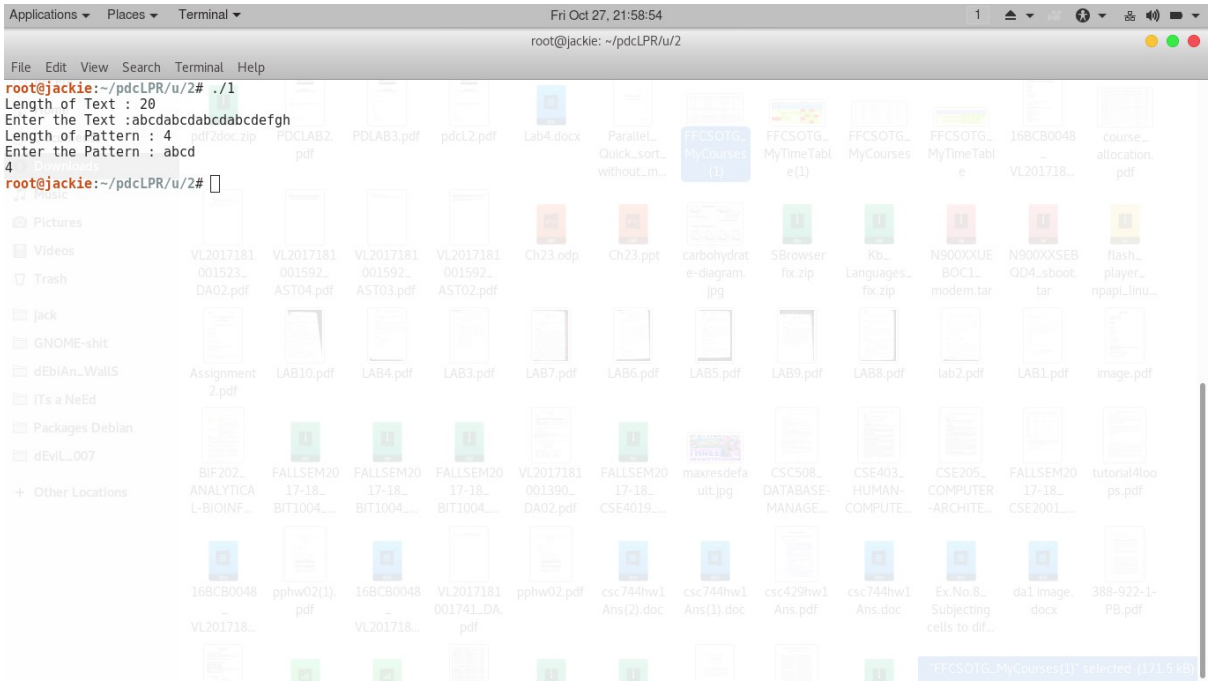
# PARALLEL AND DISTRIBUTED COMPUTING

1.The string matching problem is to find all occurrences of a particular sub-string, called the pattern, in another string, called the text. Design a parallel algorithm to solve the string matching problem using openMP.

CODE:

```
#include<iostream>
#include<omp.h>

using namespace std;
int check(char* text,char* patt,int t,int p,int j){
int s = 0,i;
```

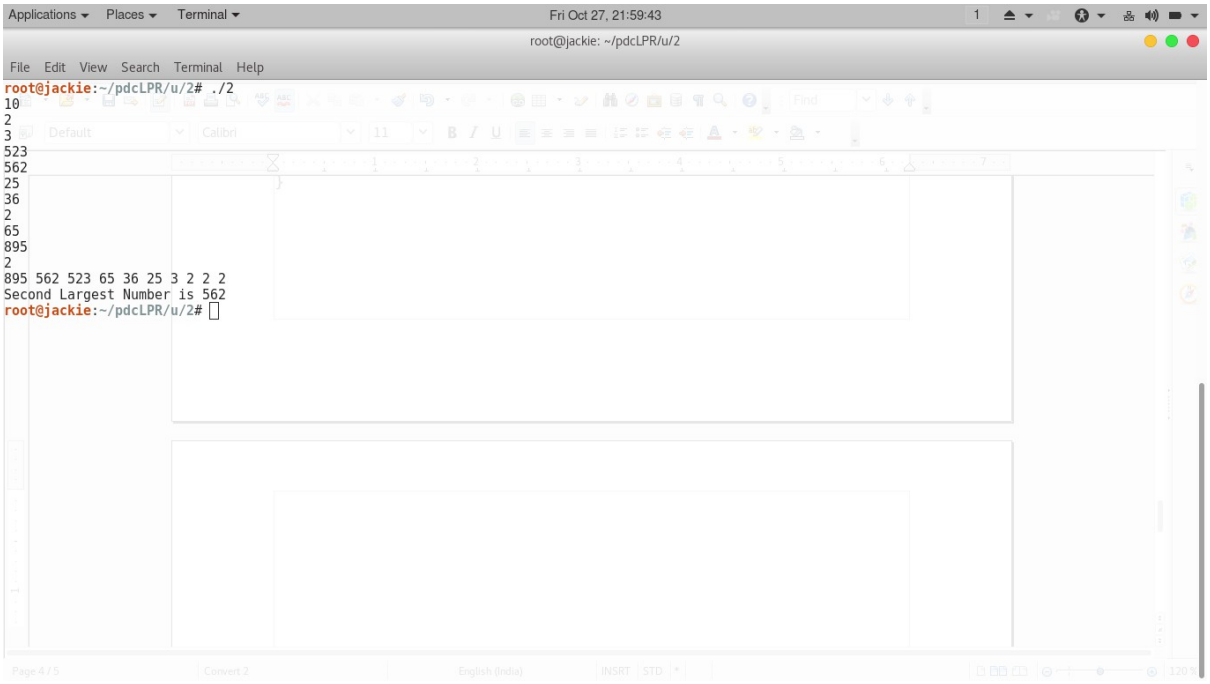
```
#pragma omp for
for(i=j;i<j+p;i++)
if(text[i]==patt[i-j])
s++;
if(s==p)
return 1;
return 0;
}

int main(){

int t,p;
cout<<"Length of Text : ";
cin>>t;
char text[t];
cout<<"Enter the Text :";
cin>>text;

cout<<"Length of Pattern : ";
cin>>p;
char patt[p];
cout<<"Enter the Pattern : ";
cin>>patt;
int start = 0,end = t-p,i,ans=0;
#pragma omp for
for(i=start;i<=end;i++)
if(check(&text[0],&patt[0],t,p,i)==1)
ans++;
cout<<ans<<endl;
return 0;
}
```


Applications
Places
Terminal
Fri Oct 27, 21:58:54
root@jackie: ~/pdcLPR/u/2
File Edit View Search Terminal Help
root@jackie:~/pdcLPR/u/2# ./1
Length of Text : 20
Enter the Text :abcdabcdabcdabcdefgh
Length of Pattern : 4
Enter the Pattern : abcd
4
root@jackie:~/pdcLPR/u/2#

2.Given a list of n keys, a[0], a[1],....a[n-1], all with distinct values , design a parallel algorithm to find the second-largest key on the list.

CODE:

```
#include<iostream>

#include<omp.h>

using namespace std;

int main(int argc, char const *argv[]) {


 int i,l,j,k;

cin>>l;

 int array[l];

 for(i=0;i<l;i++)

  cin>>array[i];


 #pragma omp
                  for

 for (i = 0; i < l;
               i++)

  for (j = i; j < l;
               j++)

    if(array[i]<array[j]){

     k = array[i];

     array[i] = array[j];

     array[j] = k;

    }

 for(i=0;i<l;i++)

  cout<<array[i]<<" ";

  cout<<endl;

 cout<<"Second Largest Number is
 "<<array[1]<<endl; return 0;

}
```


Applications
Places
Terminal
Fri Oct 27, 21:59:43
root@jackie: ~/pdcLPR/u/2
File Edit View Search Terminal Help
root@jackie:~/pdcLPR/u/2# ./2
10
2
3
523
562
25
36
2
65
895
2
895 562 523 65 36 25 3 2 2 2
Second Largest Number is 562
root@jackie:~/pdcLPR/u/2#