Fatima Jinnah Women University

Data Structure & Algorithm

LAB#01

Task 1 :

Give answers to the following.

1. Write the output of the following program.

```
#include<iostream>
using namespace std;
int mystery(int,int);
int main()
{
      int x=5,y=2;
      cout<<"Result = "<<mystery(x,y);
      return 0;
}
int mystery(int a, int b)
{
      if(b==1)
            return a;
      else
            return a + mystery(a, b-1);
}
```

Answer:

10

2. Let J and K be integers and suppose Q(J, K) is recursively defined by :

$$Q(J,K) = \begin{cases} 5, & J < K \\ Q(J-K, K+2) + J, & J \geq K \end{cases}$$

Trace and Find Q(5, 3).

Answer:

10

3. Let ‘a’ and ‘b’ be integers and suppose Q(a, b) is recursively defined by :

$$Q(a,b) = \begin{cases} 0, & a < b \\ Q(a-b,b)+1, & b \le a \end{cases}$$

Find Q(14,3).

Answer:

4

4. Identify the problem with following recursive function.

```
void recurse( int count )
{
  cout<< count <<"\n";
  recurse ( count + 1 );
}
```

In this function no condition is provided.It will be executed repeatedly for indefinite time. To fix this function, you need to add a base case that specifies when the recursion should terminate. For example, you could set a condition where the recursion stops when count reaches a certain value.

5. Given the following function, write the output if the user enters ‘abcz’ as input.

```
void rev()
{
      char c;
      cin>>c;
      if(c!='z'){
            rev();
            cout<<c;
      }
}
```

Answer:

cba

Task 2 :

Implement the following exercises.

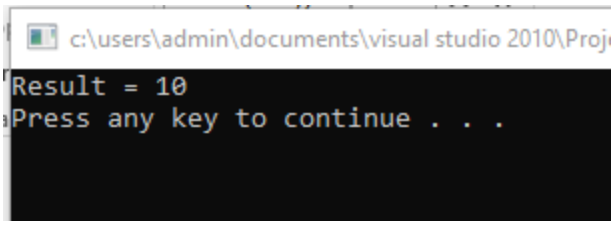
Exercise 1

Write a function sum(int a[], int size) to (recursively) compute the sum of the elements in an array.

Example Run :

```
int arr[]={1,2,3,4} ;
int result = sum(arr,4) ;
cout<<result<<endl ; //Should print 10
```

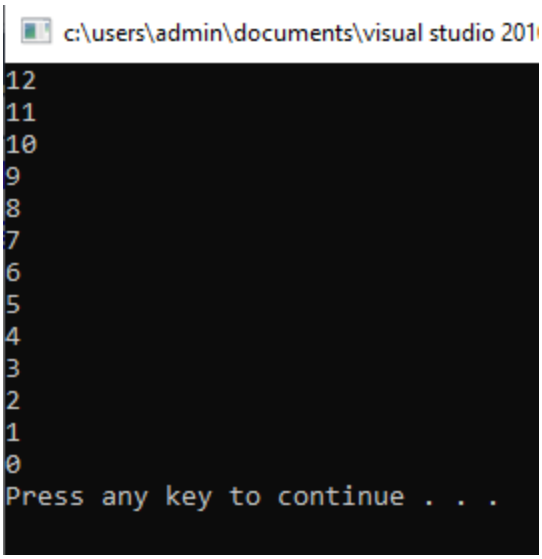
```
// 147.cpp : Defines the entry point for the console application.
//

#include "stdafx.h"
#include<iostream>
using namespace std;

int sum(int arr[] , int a);




int _tmain(int argc, _TCHAR* argv[])
{
int arr[]={1,2,3,4};
cout<<"Result = "<<sum(arr,4)<<endl;
system("pause");
return 0;

}
int sum(int arr[], int a)
{
if(a==0)
return 0;
else
return arr[a-1]+sum(arr,a-1);
}
```



Exercise 2

Write a recursive function to print integers from a given number N to 0. When called as `print (10)`, the function should print : 10 9 8 7 6 5 4 3 2 1 0

```
// 147.cpp : Defines the entry point for the console application.
//

#include "stdafx.h"
#include<iostream>
using namespace std;
void print( int a);




int _tmain(int argc, _TCHAR* argv[])
{
int a=10;
print(a);
system("pause");
return 0;

}
void print( int a)
{
if(a!=-1)
{cout<<a<<endl;
print (a-1);}

}
```



Exercise 3

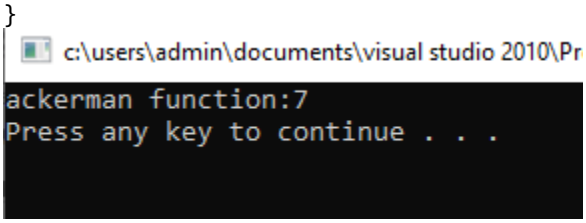
Ackermann's function is defined recursively on non-negative integers as follows.

```
A(m,n) = n+1                          if m == 0
A(m,n) = A(m-1, 1)                    if m != 0, n == 0
A(m,n) = A(m-1, A(m, n-1))            if m != 0, n != 0
```

Implement it as a recursive function Ackermann(M,N) which takes two positive integers as input and returns a positive integer as result. Once implemented test your program by evaluating Ackermann(2,2).

```
// 147.cpp : Defines the entry point for the console application.
//

#include "stdafx.h"
#include<iostream>
using namespace std;
int ackerman(int m,int n);
int _tmain(int argc, _TCHAR* argv[])
{int a=2,b=2;
cout<<"ackerman function:"<<ackerman( a, b)<<endl;
system("pause");
return 0;

}
int ackerman(int m,int n)
{
if(m==0)
return n+1;
if ((m!= 0 )&&( n== 0))
```

```
        return ackerman(m-1, 1);
if(( m!= 0) &&( n!= 0))
        return ackerman(m-1, ackerman(m, n-1));
}
```



Input

$A(2, 2)$

Result

7

Exercise 4

Binomial coefficients are normally computed using the following formula.

$$\binom{n}{m} = \frac{n!}{(n-m)!m!}$$

Binomial coefficients can also be computed using the following recursive definition.

$$\binom{n}{m} = \begin{cases} 1 & m = 0, \\ 1 & n = m, \\ \binom{n-1}{m} + \binom{n-1}{m-1} & \text{otherwise.} \end{cases}$$

Write a C++ program to compute binomial coefficients using the mentioned recursive definition.

```
// 147.cpp : Defines the entry point for the console application.
//

#include "stdafx.h"
#include<iostream>
using namespace std;
int bi(int m,int n);
int _tmain(int argc, _TCHAR* argv[])
{int a=5,b=5;
cout<<"ackerman function:"<<bi( a, b)<<endl;
system("pause");
return 0;
```

```
}
int bi(int m,int n)
{

if ((m== 0 )||( n==m ))
	return  1;
else
	return (bi(n-1),m)+bi((n-1),(m-1));
}
```

c:\users\lenovo\documents\visual studio 2010\Projects\lab 2 dsa\Deb

```
binomial of number:1
Press any key to continue . . .
```

$$\binom{n}{k} = \frac{n!}{k! \cdot (n-k)!}$$

n	6
k	6
Result	1