LAB – 2

Example problem on recursion :

1. Program to perform binary search on a set of keys.

CODE :

File Name : binary_search_function.h

```
int bin_search(int low,int high,int item,int a[])
{
int mid;
if(low>high)
return(-1);
else
{
mid=(low+high)/2;
if(item==a[mid])
return(mid);
else if(item<a[mid])
return(bin_search(low,mid-1,item,a));
else
return(bin_search(mid+1,high,item,a));
}
}
```

File Name : ex.c

```
#include <stdio.h>
#include "binary_search_function.h"
void main()
{
int i, pos, a[30],n, item;
printf("Enter number of items:");
scanf("%d",&n);
printf("Enter the elements in ascending order:\n");
for(i=0;i<n;i++)
scanf("%d",&a[i]);
printf("Enter element to be searched:");
scanf("%d",&item);
pos=bin_search(0,n-1,item,a);
if(pos!=-1)
printf("Item found at location %d",pos+1);
else
printf("Item not found");
}
```

OUTPUT :

```
$ gcc binary_search_function.h
$ gcc ex.c
$ ./a.out
Name : Manoj M Mallya
Registration number : 200905130
Batch : C2

Enter number of items:4
Enter the elements in ascending order:
2
4
6
8
Enter element to be searched:6
Item found at location 3$
$
$ ./a.out
Name : Manoj M Mallya
Registration number : 200905130
Batch : C2

Enter number of items:4
Enter the elements in ascending order:
2
23
45
67
Enter element to be searched:54
Item not found
```

PROBLEMS :

1. Create a structure STUDENT consisting of variables of structures:
i. DOB {day, month (use pointer), year},
ii. STU_INFO {reg_no, name(use pointer), address},
iii. COLLEGE {college_name (use pointer), university_name}
where structure types from i to iii are declared outside the STUDENT independently. Show how to read and display member variables of DOB type if pointer variable is created for DOB inside STUDENT and STUDENT variable is also a pointer variable. The program should read and display the values of all members of STUDENT structure.

CODE :

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
```

```
struct DOB
{
   int day;
   char *month;
   int year;
};
struct STU_INFO
{
   int reg_no;
   char *name;
   char adrs[20];
};
struct COLLEGE
{
   char *clg_name;
   char univ_name[20];
};
struct STUDENT
{
   struct DOB* dob;
   struct STU_INFO stu_info;
   struct COLLEGE clg;
};
int main()
{
   printf("Name : Manoj M Mallya\nRegistration number : 200905130\nBatch : C2\n\n");
   struct STUDENT *stu = (struct STUDENT*)malloc(sizeof(struct STUDENT));
   stu->dob = (struct DOB*)malloc(sizeof(struct DOB));
   stu->dob->month = (char*)malloc(sizeof(char) * 10);
   stu->stu_info.name = (char*)malloc(sizeof(char) * 20);
   stu->clg.clg_name = (char*)malloc(sizeof(char) * 50);
   printf("Enter Details of Student : \n");
   printf("Registration Number : ");
   scanf("%d", &(stu->stu_info.reg_no));
   printf("Name : ");
   scanf(" %[^\n]s", stu->stu_info.name);
   printf("Address : ");
   scanf(" %[^\n]s", stu->stu_info.adrs);
   printf("Date of Birth (DD MONTH YYYY) : ");
   scanf("%d", &(stu->dob->day));
   scanf(" %s", stu->dob->month);
   scanf("%d", &(stu->dob->year));
   printf("College Name : ");
   scanf(" %[^\n]s", stu->clg.clg_name);
   printf("University Name : ");
   scanf(" %[^\n]s", stu->clg.univ_name);
   printf("\nStudent Details : \n\nRegistration Number : %d\nName : %s\nAddress : %s\nDateof
Birth : %d %s %d\nCollege Name : %s\nUniversity Name : %s\n\n", stu->stu_info.reg_no, stu-
>stu_info.name, stu->stu_info.adrs,stu->dob->day,stu->dob->month,stu->dob->year,stu-
>clg.clg_name,stu->clg.univ_name);
   return 0;
}
```

OUTPUT :

```
$ gcc 2_1.c
$ ./a.out
Name : Manoj M Mallya
Registration number : 200905130
Batch : C2

Enter Details of Student :
Registration Number : 135798642
Name : Shashank Balaji
Address : KM nagara, Shivamogga
Date of Birth (DD MONTH YYYY) : 13 February 2000
College Name : Sharada PU college
University Name : Indian Institute of Technology,Mumbai

Student Details :

Registration Number : 135798642
Name : Shashank Balaji
Address : KM nagara, Shivamogga
Dateof Birth : 13 February 2000
College Name : Sharada PU college
University Name : Indian Institute of Technology,Mumbai
```

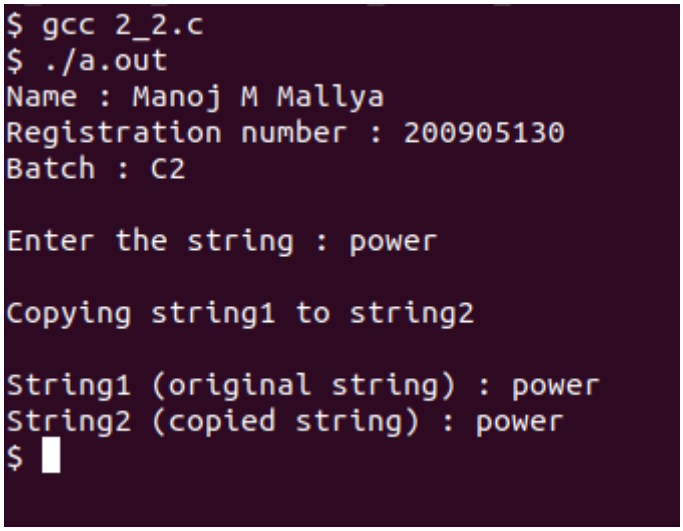
2. Write C programs using recursion to copy one string to another using Recursion.

CODE :

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>

void copy(char str1[], char str2[], int i)
{
   str2[i]=str1[i];
   if(str1[i] == '\0')
      return;
   copy(str1,str2,++i);
}

int main()
{
   printf("Name : Manoj M Mallya\nRegistration number : 200905130\nBatch : C2\n\n");
   char str1[100],str2[100];
   printf("Enter the string : ");
   scanf("%[^\n]s",str1);
   printf("\nCopying string1 to string2\n");
   copy(str1,str2,0);
```

```
    printf("\nString1 (original string) : %s\n",str1);
    printf("String2 (copied string) : %s\n",str2);
    return 0;
}
```

OUTPUT :



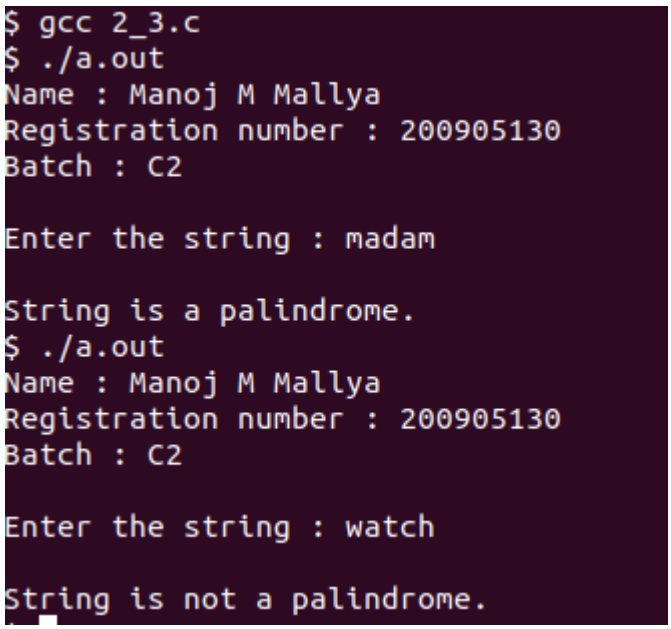
3. Write C programs using recursion to check whether a given String is Palindrome or not, using Recursion.

CODE :

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

int palindromecheck(char str[],int i,int j)
{
    if(j>i)
    {
        if(str[i]==str[j])
        {
            return palindromecheck(str,i+1,j-1);
        }
        else
        {
            return 0;
        }
    }
    return 1;
}

int main()
{
    printf("Name : Manoj M Mallya\nRegistration number : 200905130\nBatch : C2\n\n");
```

```
    char str[100];
    printf("Enter the string : ");
    scanf(" %[^\n]s",str);

    int l = strlen(str);
    int c = palindromecheck(str,0,l-1);
    if(c == 1)
    {
        printf("\nString is a palindrome.\n");
    }
    else
    {
        printf("\nString is not a palindrome.\n");
    }
    return 0;
}
```

OUTPUT :



4. Write C programs using recursion to simulate the working of Tower of Hanoi for n disks. Print the number of moves.

CODE :

```
#include <stdio.h>
#include <stdlib.h>

int tower(int n,char s,char t,char d)
{
    static int count = 0;
    count++;
```

```
    if (n==1)
    {
        printf("Moving disc from %c to %c\n",s,d);
        return count;
    }
    tower(n-1,s,d,t);
    printf("Moving disc from %c to %c\n",s,d);
    tower(n-1,t,s,d);
}

int main()
{
    printf("Name : Manoj M Mallya\nRegistration number : 200905130\nBatch : C2\n\n");
    int n,m;
    printf("Enter the number of disks used for tower of hanoi : ");
    scanf("%d",&n);

    char s='a',d='c',t='b'; //s - source , d - destination , t - temporary
    m = tower (n,s,t,d);
    printf("\nThe number of moves taken : %d\n",m);
    return 0;
}
```

OUTPUT :

```
$ gcc 2_4.c
$ ./a.out
Name : Manoj M Mallya
Registration number : 200905130
Batch : C2

Enter the number of disks used for tower of hanoi : 3
Moving disc from a to c
Moving disc from a to b
Moving disc from c to b
Moving disc from a to c
Moving disc from b to a
Moving disc from b to c
Moving disc from a to c

The number of moves taken : 7
```

****