

LAB-2

Q1. Write a recursive program to find the sum of n integers

```
Ans. import java.util.*;
public class Q1
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter n:");
        int n = sc.nextInt();
        int sum = rsum(n);
        System.out.println("Sum is:" + sum);
    }
    public static int rsum(int n)
    {
        if (n == 0)
        {
            return 0;
        }
        else if (n == 1)
        {
            Scanner sc = new Scanner(System.in);
            int y = sc.nextInt();
            return (y);
        }
    }
}
```

```
else  
{  
    Scanner sc = new Scanner(System.in);  
    int y = sc.nextInt();  
    return (sum(n-1) + y);  
}  
}  
}
```

Output:

Enter n: 4

2

4

8

1

Sum is 15.

Q2. Write a recursive program to find the min^m element from an array of n input.

Ans.

```
import java.util.*;
```

```
public class Q2
```

```
{
```

```
    public static void main (String[] args)
```

```
    {
```

```
        Scanner sc = new Scanner (System.in);
```

```
        System.out.print ("Enter no. of term: ");
```

```
        int n = sc.nextInt();
```

```
        int a[] = new int[n];
```

```
        for (int i = 0; i < a.length; i++)
```

```
        { a[i] = sc.nextInt();
```

```
        }
```

```
        System.out.println();
```

```
        int sm = rmin(a, n-1);
```

```
        System.out.println(sm) ("Minimum no. is" + sm);
```

```
    }
```

```
    public static int rmin (int a, int b)
```

```
    {
```

```
        if (a < b)
```

```
        { return a;
```

```
        }
```

```
        else
```

```
        { return b;
```

```
        }
```

```
    }
```

```
public static int rmin(int a[], int n)
{
    if (n <= 1)
    {
        return a[0];
    }
    else
    {
        return (min(rmin(a, n-1), a[n]));
    }
}
```

Output:

Enter no. of term: 5

4

12

3

1

6

Minimum no. is 1.

Q3: Write a recursive program to find the factorial of a no.

Ans.

```
import java.util.*;  
public class Q3  
{  
    public static void main (String[] args)  
    {  
        Scanner sc = new Scanner (System.in);  
        System.out.print ("Enter n: ");  
        int n = sc.nextInt();  
        int fact = fact(n);  
        System.out.println ("Factorial of " + n + " is "  
            + f);  
    }  
    public static int fact (int n)  
    {  
        if (n == 0 || n == 1)  
        {  
            return (1);  
        }  
        else  
        {  
            return (fact(n-1) * n);  
        }  
    }  
}
```

Output : Enter n : 5
Factorial of 5 is 120.

Q4. Write a recursive program to generate the n^{th} Fibonacci no.

Ans.

```
import java.util.*;  
public class Q4  
{  
    public static void main(String[] args)  
    {  
        Scanner sc = new Scanner(System.in);  
        System.out.print("Enter the term position  
to display: ");  
        int n = sc.nextInt();  
        int res = fbn(n);  
        System.out.println("The no. is " + res);  
    }  
    public static int fbn(int n)  
    {  
        if (n == 1)  
        {  
            return 0;  
        }  
        else if (n == 2)  
        {  
            return 1;  
        }  
        else  
        {  
            return (fbn(n-1) + fbn(n-2));  
        }  
    }  
}
```

Output: Enter the term posⁿ to display: 8
The no. is 13.

Q5: Write a recursive program to ~~generate~~ find search an element from an array of a input.

Ans.

```
import java.util.*;
```

```
public class Q4
```

```
{  
    public static void main (String[] args)
```

```
{
```

```
    Scanner sc = new Scanner (System.in);
```

```
    System.out.print ("Enter no. of term: ");
```

```
    int n = sc.nextInt();
```

```
    int a[] = new int[n];
```

```
    for (int i = 0; i < a.length; i++)
```

```
    {  
        a[i] = sc.nextInt();
```

```
    }
```

```
    System.out.println();
```

```
    System.out.println ("Enter elements to be  
searched:");
```

```
    int ele = sc.nextInt();
```

```
    int x = search(a, n-1, ele);
```

```
    if (x >= 0)
```

```
    {  
        System.out.println ("Found");
```

```
    }
```

```
    else
```

```
    {
```

```
        System.out.println ("Not found");
```

```
    }
```

```
}
```

```
public static int search (int a[], int n, int ele)
{
    if (n == 0)
    {
        return (-1);
    }
    else
    {
        if (a[n] == ele)
        {
            return (n);
        }
        else
        {
            return (search(a, n-1, ele));
        }
    }
}
```

Output: Enter no. of term: 5

8

4

3

1

2

Enter element to be searched: 9.

Not found!

Q6. Using binary search, write a ~~program~~ recursive program to search an element from array

Ans.

```
import java.util.*;  
public class Q6  
{  
    public static void main (String[] args)  
    {  
        Scanner sc = new Scanner (System.in);  
        System.out.print ("Enter no. of terms: ");  
        int n = sc.nextInt();  
        int a[] = new int [n];  
        for (int i = 0; i < a.length; i++)  
        {  
            a[i] = sc.nextInt();  
        }  
        System.out.println();  
        System.out.print ("Enter no. to search: ");  
        int ele = sc.nextInt();  
        int start = 0;  
        int end = n - 1;  
        int res = search (a, start, end, ele);  
        if (res >= 0)  
        {  
            System.out.println ("Found");  
        }  
        else  
        {  

```

```
        System.out.println("Not found");
    }
}

public static int rebsrch (int a[], int s, int e,
int ele)
{
    int mid = (s+e)/2
    if (a[mid] == ele)
    {
        return mid;
    }
    else if (ele < a[mid])
    {
        return rebsrch (a, s, (mid-1), ele);
    }
    else
    {
        return rebsrch (a, (mid+1), e, ele);
    }
}
}
```

Output: Enter no. of terms : 5

4
8
2
1
3

Enter no. to search : 3 .

Found!

Q7. Write a recursive program, to compute x^n .

Ans

```
import java.util.*;
```

```
public class Q7
```

```
{ public static void main (String[] args)
{
```

```
    Scanner sc = new Scanner (System.in);
```

```
    System.out.print ("Enter base :");
```

```
    int x = sc.nextInt();
```

```
    System.out.print ("Enter power :");
```

```
    int n = sc.nextInt();
```

```
    int res = pow (x, n);
```

```
    System.out.println (x + " to the power " + n
+ " is " + res);
```

```
}
```

```
public static int pow (int b, int p)
```

```
{ if (p == 0)
```

```
{ return 1;
```

```
}
```

```
else
```

```
{ return (b * pow (b, p-1));
```

```
}
```

```
}
```

Output : Enter x : 3

Enter power : 4

3 to the power 4 is 81

Q8. Write a recursive program to find the sum of the digits of a no. n .

Ans.

```
import java.util.*;  
public class Q8  
{  
    public static void main (String[] args)  
    {  
        Scanner sc = new Scanner (System.in);  
        System.out.print ("Enter n : ");  
        int n = sc.nextInt();  
        int sum = 0;  
        int res = digitsum (n, sum);  
        System.out.println ("Sum of the digit is : "  
        + res);  
    }  
    public static int digitsum (int n, int sum)  
    {  
        int if (n == 0)  
        {  
            return sum;  
        }  
        else  
        {  
            sum += n % 10;  
            return (digitsum (n / 10, sum));  
        }  
    }  
}
```

Output : Enter n : 213
Sum of the digit is 6