

Soru 1: Main fonksiyonu içerisinde verilen bir doğal sayıya kadar olan (bu sayı dahil) tüm sayıların toplamını rekürsif bir fonksiyon kullanarak hesaplayınız.

Calculate the sum of all numbers (including this number) up to a natural number given in the main function using a recursive function.

Çözüm (5 Dakika):

Example: Sum of Natural Numbers Using Recursion

```
#include <stdio.h>
int sum(int n);

int main()
{
    int number, result;

    printf("Enter a positive integer: ");
    scanf("%d", &number);

    result = sum(number);

    printf("sum=%d", result);
}

int sum(int num)
{
    if (num!=0)
        return num + sum(num-1); // sum() function calls itself
    else
        return num;
}
```

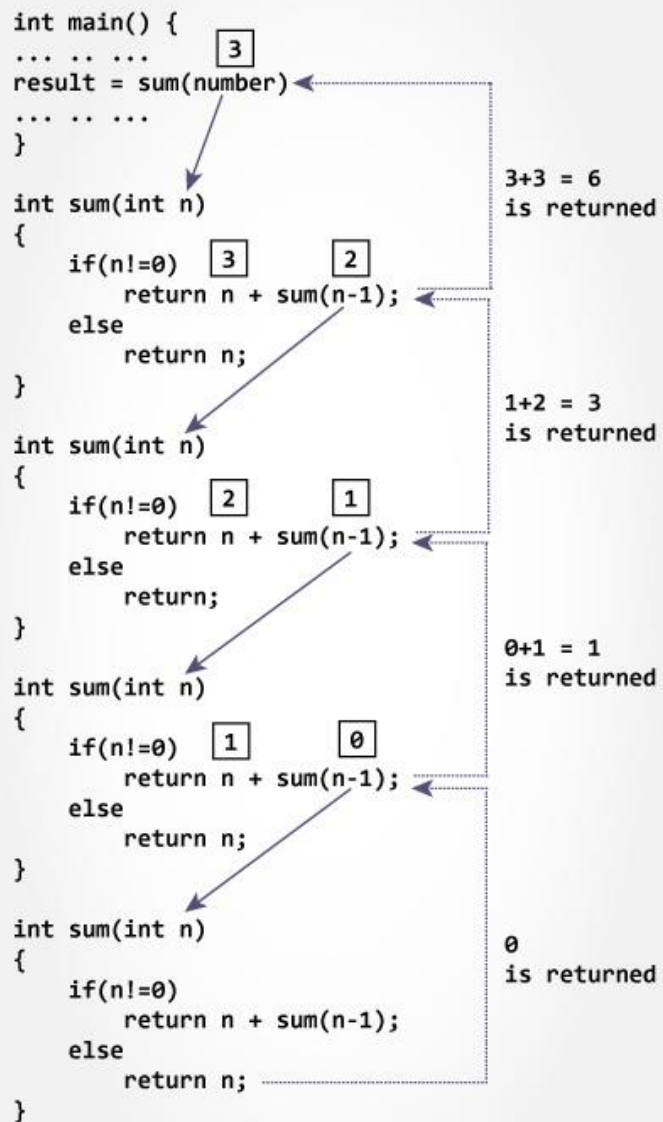
Output

```
Enter a positive integer:
3
6
```

Initially, the `sum()` is called from the `main()` function with `number` passed as an argument.

Suppose, the value of `num` is 3 initially. During next function call, 2 is passed to the `sum()` function. This process continues until `num` is equal to 0.

When `num` is equal to 0, the if condition fails and the else part is executed returning the sum of integers to the `main()` function.



Soru 3:

Girilen dizideki en büyük elemanı bulan programı C dilinde Rekürsif olarak yazınız.

Write a recursion program that finds the largest element in the entered array in C language.

```
#include<stdio.h>
#define MAX 100

int getMaxElement(int []); // takes array of int as parameter
int size;

int main(){
    printf("\n\n\t\tStudytonight - Best place to learn\n\n\n");
    int arr[MAX], max, i;
    printf("\n\nEnter the size of the array: ");
    scanf("%d", &size);
    printf("\n\nEnter %d elements\n\n", size);

    for(i = 0; i < size; i++)
    {
        scanf("%d", &arr[i]);
    }

    max = getMaxElement(arr); // passing the complete array as parameter
    printf("\n\nLargest element of the array is %d\n\n", max);
    printf("\n\n\t\t\tCoding is Fun !\n\n\n");
    return 0;
}

int getMaxElement(int a[]){
    static int i = 0, max = -9999; // static int max=a[0] is invalid
    if(i < size) // till the last element
    {
        if(max < a[i])
            max = a[i];

        i++; // to check the next element in the next iteration
        getMaxElement(a); // recursive call
    }
    return max;
}
```

Soru 3:

Main fonksiyonu içerisinde verilen sayının basamaklarını toplamını rekürsif bir fonksiyon kullanarak bulunuz.

Write a recursion function to sum the digits of the number given in the main function.

Çözüm (5 Dakika):

```
1. /*
2.  * C Program to find Sum of Digits of a Number using Recursion
3.  */
4. #include <stdio.h>
5.
6. int sum (int a);
7.
8. int main()
9. {
10.     int num, result;
11.
12.     printf("Enter the number: ");
13.     scanf("%d", &num);
14.     result = sum(num);
15.     printf("Sum of digits in %d is %d\n", num, result);
16.     return 0;
17. }
18.
19. int sum (int num)
20. {
21.     if (num != 0)
22.     {
23.         return (num % 10 + sum (num / 10));
24.     }
25.     else
26.     {
27.         return 0;
28.     }
29. }
```

Soru 4:

Main fonksiyonu içerisinde verilen iki pozitif sayıyı kullanarak bu sayıların en büyük ortak bölenini rekürsif bir fonksiyon kullanarak bulunuz.

Using the two positive numbers given in the main function, find the largest common divisor of these numbers using a recursive function.

Çözüm (5 Dakika):

Example: GCD of Two Numbers using Recursion

```
#include <stdio.h>
int hcf(int n1, int n2);
int main()
{
    int n1, n2;
    printf("Enter two positive integers: ");
    scanf("%d %d", &n1, &n2);

    printf("G.C.D of %d and %d is %d.", n1, n2, hcf(n1,n2));
    return 0;
}

int hcf(int n1, int n2)
{
    if (n2 != 0)
        return hcf(n2, n1%n2);
    else
        return n1;
}
```

Output

```
Enter two positive integers: 366
60
G.C.D of 366 and 60 is 6.
```