

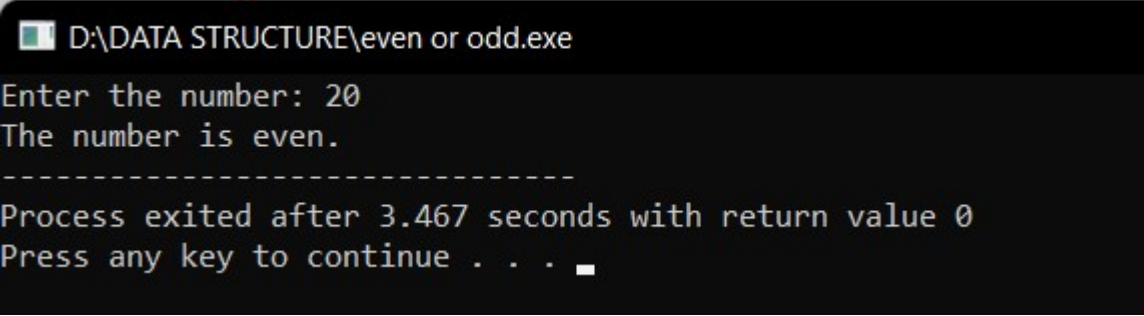
25/08/2023

PROGRAM1: To find given number is even or odd

Code:

```
#include<stdio.h>
int main()
{
    int n;
    printf("Enter the number: ");
    scanf("%d",&n);
    if(n%2==0)
        printf("The number is even.");
    else
        printf("The number is odd.");
}
```

Sample input and Output:



PROGRAM2: To find sum of first n numbers using any loop

Code:

```
#include <stdio.h>
int main()
{
    int i, n, sum=0;
    printf("Enter the limit: ");
```

```

scanf("%d", &n);
for(i=1; i<=n; i++)
{
    sum = sum + i;

printf("Sum of first %d natural numbers = %d", n, sum);
}

```

Sample input and Output:



PROGRAM3: To find sum of first n even numbers

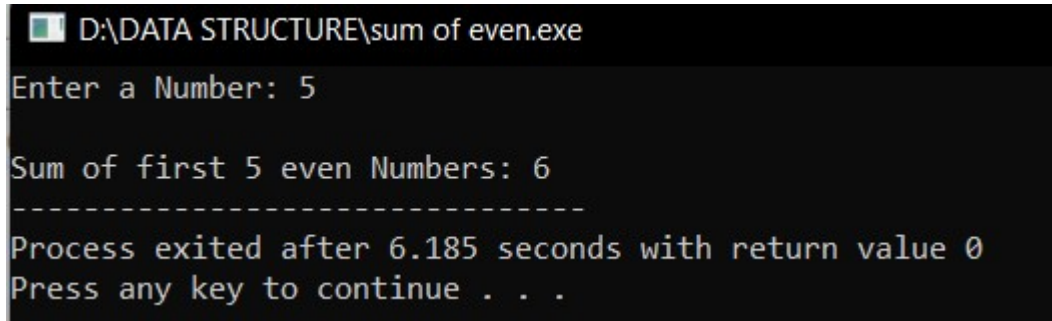
Code:

```

#include<stdio.h>
int main()
{
    int n, i, sum = 0 ;
    printf("Enter a Number: ");
    scanf("%d",&n);
    for(i=1;i<=n;i++)
    {
        if(i%2==0)
        {
            sum = sum + i;
        }
    }
    printf("\nSum of first %d even Numbers: %d",n,sum);
}

```

Sample input and Output:



```
D:\DATA STRUCTURE\sum of even.exe
Enter a Number: 5

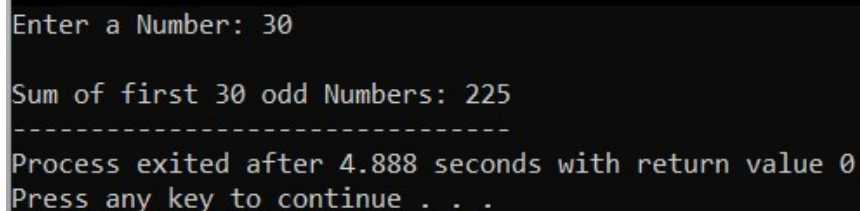
Sum of first 5 even Numbers: 6
-----
Process exited after 6.185 seconds with return value 0
Press any key to continue . . .
```

PROGRAM4: To find sum of first n odd numbers

Code:

```
#include<stdio.h>
int main()
{
    int n, i, sum = 0 ;
    printf("Enter a Number: ");
    scanf("%d",&n);
    for(i=1;i<=n;i++)
    {
        if(i%2!=0)
        {
            sum = sum + i;
        }
    }
    printf("\nSum of first %d even Numbers: %d",n,sum);
}
```

Sample input and Output:



```
Enter a Number: 30

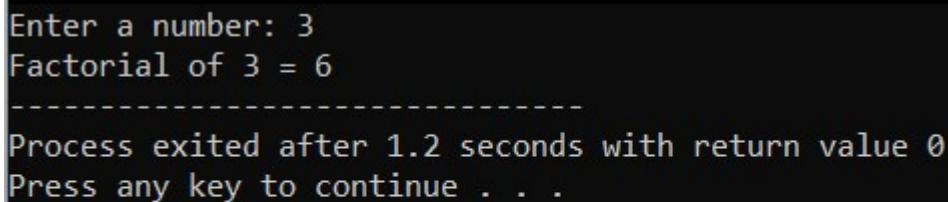
Sum of first 30 odd Numbers: 225
-----
Process exited after 4.888 seconds with return value 0
Press any key to continue . . .
```

PROGRAM5: To find factorial of given number with recursion

Code:

```
#include<stdio.h>
int multiplyNumbers(int n);
int main() {
    int n;
    printf("Enter a number: ");
    scanf("%d",&n);
    printf("Factorial of %d = %ld", n, multiplyNumbers(n));
    return 0;
}
int multiplyNumbers(int n) {
    if (n>=1)
        return n*multiplyNumbers(n-1);
    else
        return 1;
}
```

Sample input and Output:

A screenshot of a terminal window showing the execution of the factorial program. The user enters '3' when prompted. The program outputs 'Factorial of 3 = 6'. Below this, a dashed line separates the output from the program's status message: 'Process exited after 1.2 seconds with return value 0'. The prompt 'Press any key to continue . . .' is shown at the bottom.

```
Enter a number: 3
Factorial of 3 = 6
-----
Process exited after 1.2 seconds with return value 0
Press any key to continue . . .
```

PROGRAM6: To find factorial of given number

Code:

```
#include <stdio.h>
int main() {
    int n, i;
    int fact = 1;
    printf("Enter a number: ");
    scanf("%d", &n);
```

```

if (n < 0)
    printf("Error! Factorial of a negative number doesn't exist.");
else {
    for (i = 1; i <= n; ++i) {
        fact = fact*i;
    }
    printf("Factorial of %d = %d", n, fact);
}
}

```

Sample input and Output:

```

Enter a number: 5
Factorial of 5 = 120
-----
Process exited after 3.227 seconds with return value 0
Press any key to continue . . .

```

PROGRAM7: To find fibonacci series with recursion

Code:

```

#include <stdio.h>
int fib(int a, int b, int sum, int N)
{
    if (N != 0) {
        printf(" %d", a);
        sum = a + b;
        a = b;
        b = sum;
        N--;
        fib(a, b, sum, N);
    }
}
int main()
{
    int n;
    printf("Enter till what number: ");
    scanf("%d",&n);
    fib(0, 1, 0, n);
    return 0;
}

```

Sample input and Output:

```
Enter the limit of a number : 10
0 1 1 2 3 5 8 13 21 34
-----
Process exited after 4.362 seconds with return value 0
Press any key to continue . . .
```

PROGRAM8: To find fibonacci series

Code:

```
    printf("%d, ", nextTerm);
    t1 = t2;
    t2 = nextTerm;
    nextTerm = t1 + t2;
}
return 0;#include <stdio.h>
int main() {
    int t1 = 0, t2 = 1, nextTerm = 0, n;
    printf("Enter a number: ");
    scanf("%d", &n);
    printf("Fibonacci Series: %d, %d, ", t1, t2);
    nextTerm = t1 + t2;
    while (nextTerm <= n) {

    }
```

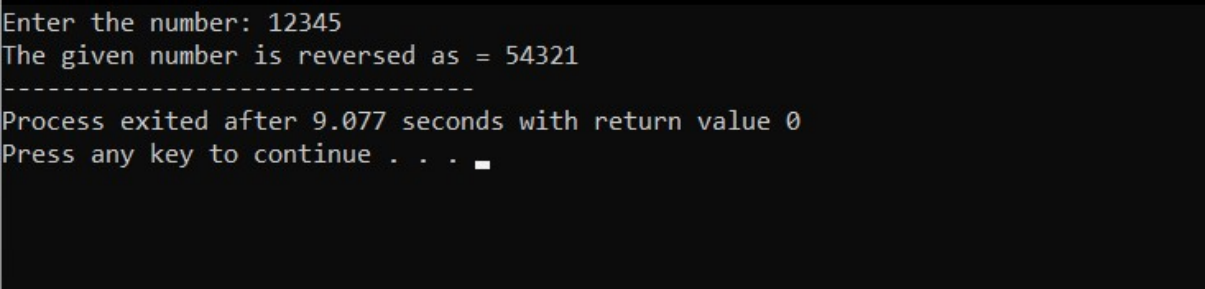
Sample input and Output:

```
Enter the number: 10
Fibonacci Series: 0, 1, 1, 2, 3, 5, 8,
-----
Process exited after 1.38 seconds with return value 0
Press any key to continue . . .
```

PROGRAM9: To reverse a given number

Code:

```
#include <stdio.h>
int main() {
    int n, rev = 0, rem;
    printf("Enter a number: ");
    scanf("%d", &n);
    while (n != 0) {
        rem = n % 10;
        rev = rev * 10 + rem;
        n = n/10;
    }
    printf("Reversed number = %d", rev);
    return 0;
}
```

Sample input and Output:

```
Enter the number: 12345
The given number is reversed as = 54321
-----
Process exited after 9.077 seconds with return value 0
Press any key to continue . . .
```

PROGRAM10: To check whether number is palindrome or not

Code:

```
#include <stdio.h>
int main() {
    int n, rev = 0, rem, temp;
    printf("Enter a number: ");
    scanf("%d", &n);
    temp = n;
    while (n != 0) {
        rem = n % 10;
        rev = rev * 10 + rem;
        n /= 10;
    }
}
```

```

    if (temp == rev)
        printf("%d is a palindrome.", temp);
    else
        printf("%d is not a palindrome.", temp);

    return 0;
}

```

Sample input and Output:

```

Enter a number: 1991
1991 is a palindrome
-----
Process exited after 11.88 seconds with return value 0
Press any key to continue . . .

```

PROGRAM11: To check whether number is armstrong or not

Code:

```

#include <stdio.h>
int main() {
    int n, ori, rem, result = 0;
    printf("Enter a number: ");
    scanf("%d", &n);
    ori = n;
    while (ori != 0) {
        rem = ori % 10;
        result += rem * rem * rem;
        ori /= 10;
    }
}

```



```
if (result == n)
    printf("%d is an Armstrong number.", n);
else
    printf("%d is not an Armstrong number.", n);
return 0;
}
```

Sample input and Output:

```
enter the number=153
armstrong number
-----
Process exited after 1.304 seconds with return value 0
Press any key to continue . . . █
```