K.R. MANGALAM UNIVERSITY
THE COMPLETE WORLD OF EDUCATION

Course: Data Structure

(Course Code : ENCS205)

UNIT-1: Foundations of Data Structures

School of Engineering & Technology
K.R. Mangalam University

SESSION 7:

Analysis of Iterative and Recurrence Relations

Recapitulation (Previous Session)

Quiz

Q1: What is a recurrence relation in the context of algorithms and data structures?
a) A relation between different data structures
b) A relation between two recursive functions
c) A relation between elements of an array
d) A relation describing a function in terms of its own values at smaller inputs

Q2: What is the purpose of solving recurrence relations in algorithms?
a) To identify the runtime complexity of algorithms
b) To determine the output of recursive functions
c) To optimize the space complexity of data structures
d) To compare different sorting algorithms

Recapitulation (Previous Session)

Quiz

Q3: What does the solution to a recurrence relation provide?

a) The best-case scenario for algorithm runtime

b) The worst-case scenario for algorithm runtime

c) The average-case scenario for algorithm runtime

d) The exact number of iterations required by a loop

Q4: What is the time complexity of an algorithm described by the recurrence relation T(n) = 2T(n/2) + n?

a) O(n)

b) O(n log n)

c) O(n^2)

d) O(log n)

Recapitulation (Previous Session)

Quiz (Answers)

A1: d) A relation describing a function in terms of its own values at smaller inputs.

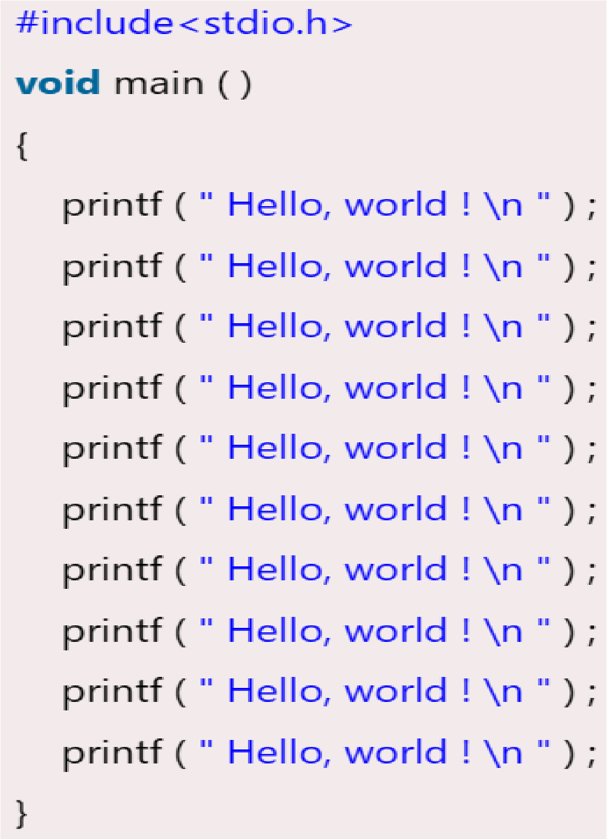
A2: a) To identify the runtime complexity of algorithms

A3: b) The worst-case scenario for algorithm runtime

A4: b) O(n log n)

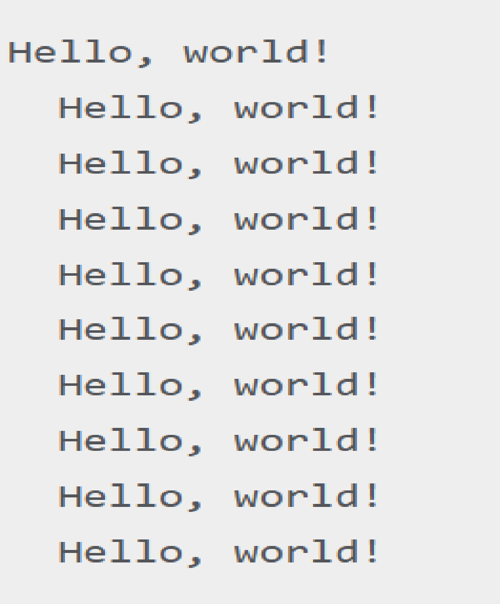
Iterative Relations

Program to print the " Hello world " 10 times in C language without using iterations.



```c
#include<stdio.h>
void main ( )
{
    printf ( " Hello, world ! \n " ) ;
    printf ( " Hello, world ! \n " ) ;
    printf ( " Hello, world ! \n " ) ;
    printf ( " Hello, world ! \n " ) ;
    printf ( " Hello, world ! \n " ) ;
    printf ( " Hello, world ! \n " ) ;
    printf ( " Hello, world ! \n " ) ;
    printf ( " Hello, world ! \n " ) ;
    printf ( " Hello, world ! \n " ) ;
    printf ( " Hello, world ! \n " ) ;
}
```

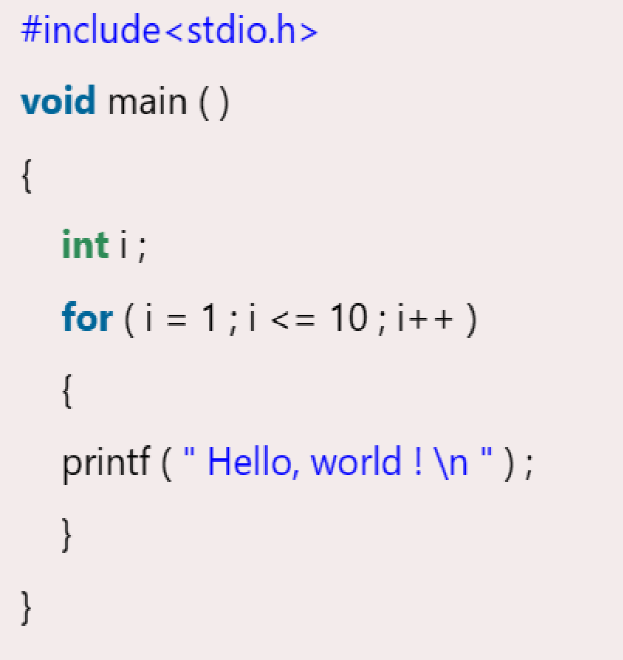
Output



```
Hello, world!
  Hello, world!
  Hello, world!
  Hello, world!
  Hello, world!
  Hello, world!
  Hello, world!
  Hello, world!
  Hello, world!
  Hello, world!
```

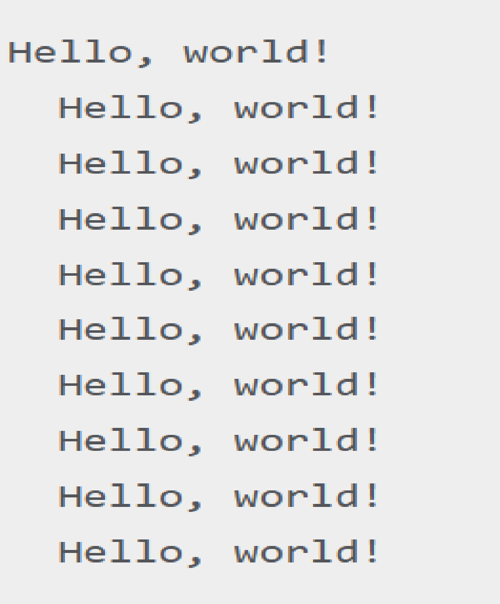
Iterative Relations

Program to print the " Hello world " 10 times in C language with using iterations



```
#include<stdio.h>
void main ( )
{
   int i ;
   for ( i = 1 ; i <= 10 ; i++ )
   {
   printf ( " Hello, world ! \n " ) ;
   }
}
```

Output



```
Hello, world!
   Hello, world!
   Hello, world!
   Hello, world!
   Hello, world!
   Hello, world!
   Hello, world!
   Hello, world!
   Hello, world!
   Hello, world!
```

Iterative Relations

- Repeating a certain number of steps continuously until a particular condition is met successfully.
- Iterations can be an infinite number of times.
- An iterative relation is a mathematical expression that defines a sequence or series by specifying how each term depends on previous terms.

Methods to perform the iterations into the program

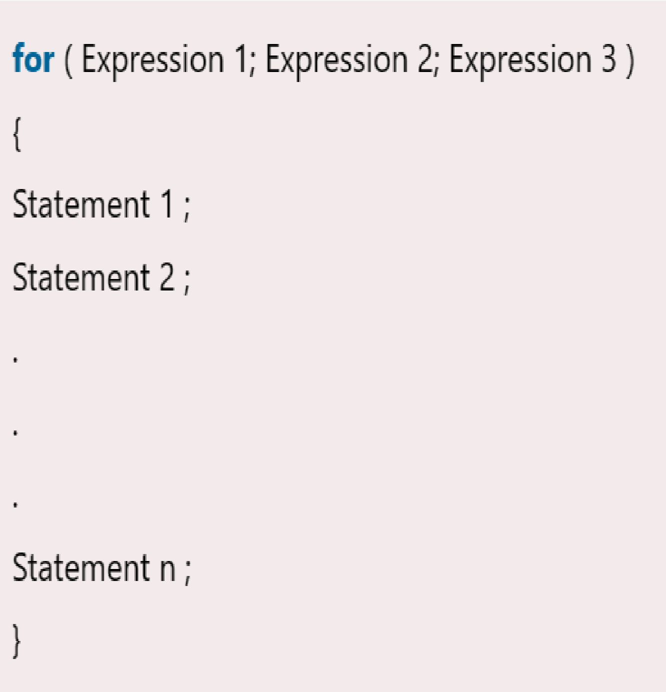
1. Using various loops in the program:

- Using For loop
- Using while loop
- Using do-while loop

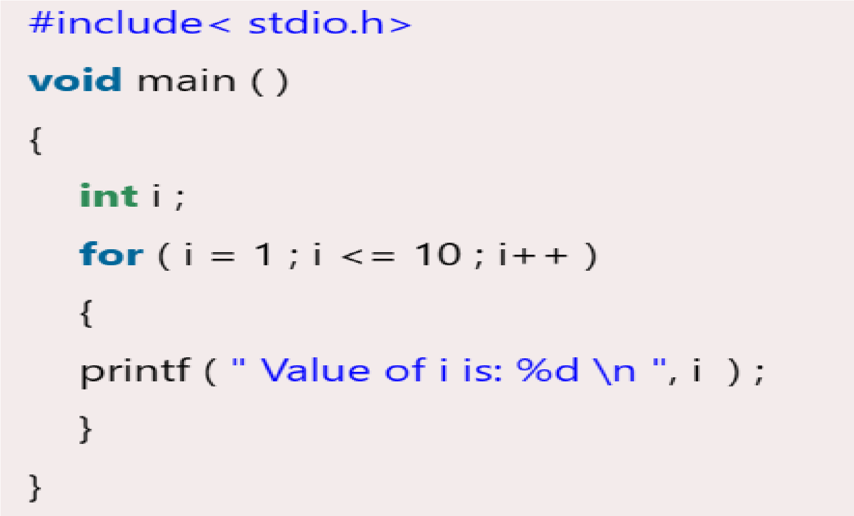
2. Using recursion

1. Using various loops in the program:

1. Using For loop



```
for ( Expression 1; Expression 2; Expression 3 )
{
Statement 1 ;
Statement 2 ;
.
.
.
Statement n ;
}
```

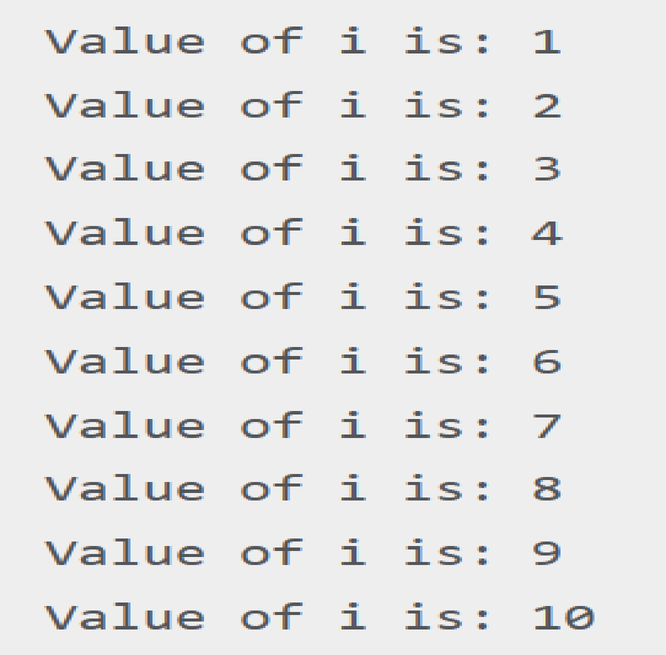


```
#include< stdio.h>
void main ( )
{
    int i ;
    for ( i = 1 ; i <= 10 ; i++ )
    {
    printf ( " Value of i is: %d \n ", i  ) ;
    }
}
```

1. Using various loops in the program:

1. Using For loop

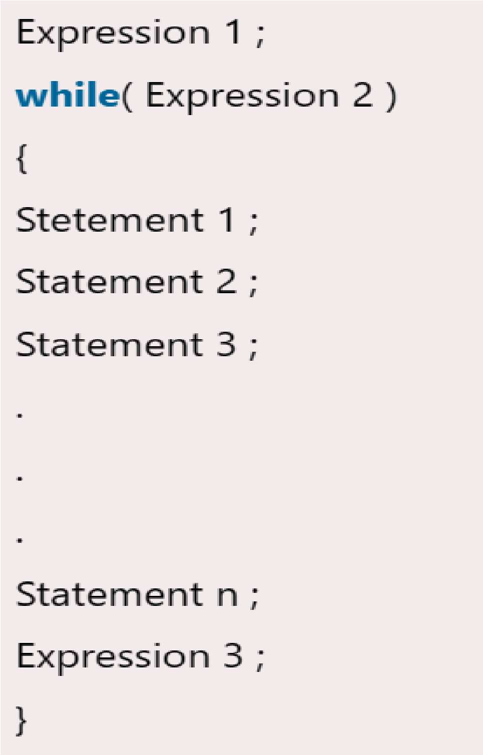
Output



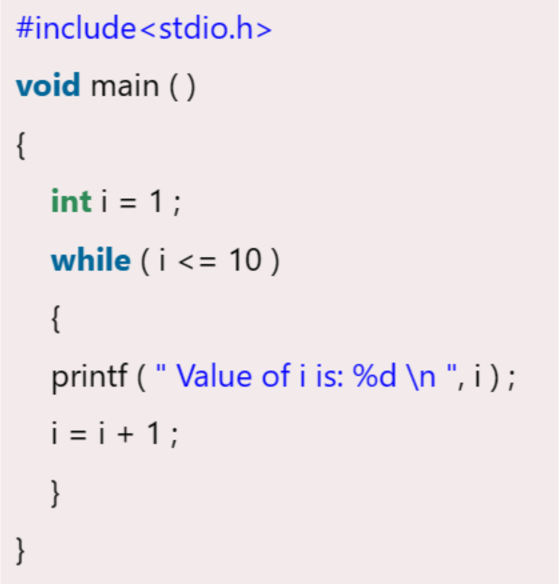
```
Value of i is: 1
Value of i is: 2
Value of i is: 3
Value of i is: 4
Value of i is: 5
Value of i is: 6
Value of i is: 7
Value of i is: 8
Value of i is: 9
Value of i is: 10
```

1. Using various loops in the program:

2. Using While loop:



```
Expression 1 ;
while( Expression 2 )
{
Stetement 1 ;
Statement 2 ;
Statement 3 ;
.
.
.
Statement n ;
Expression 3 ;
}
```

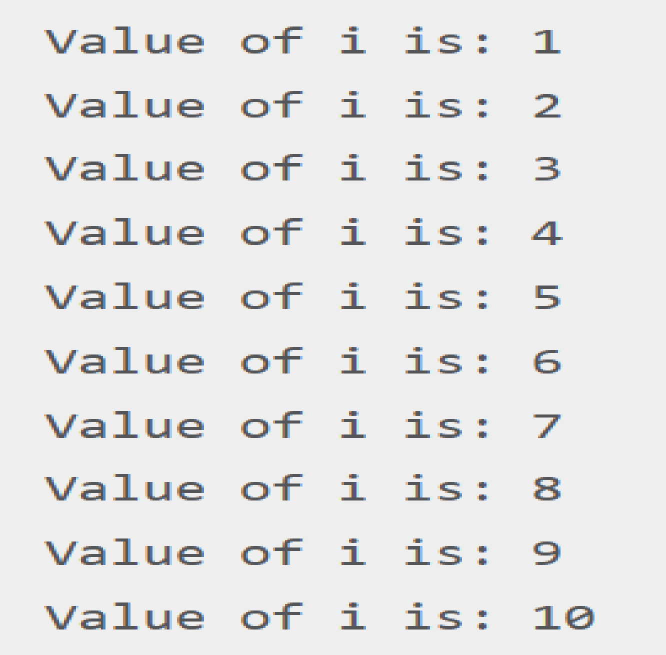


```
#include<stdio.h>
void main ( )
{
   int i = 1 ;
   while ( i <= 10 )
   {
   printf ( " Value of i is: %d \n ", i ) ;
   i = i + 1 ;
   }
}
```

1. Using various loops in the program:

2. Using While loop:

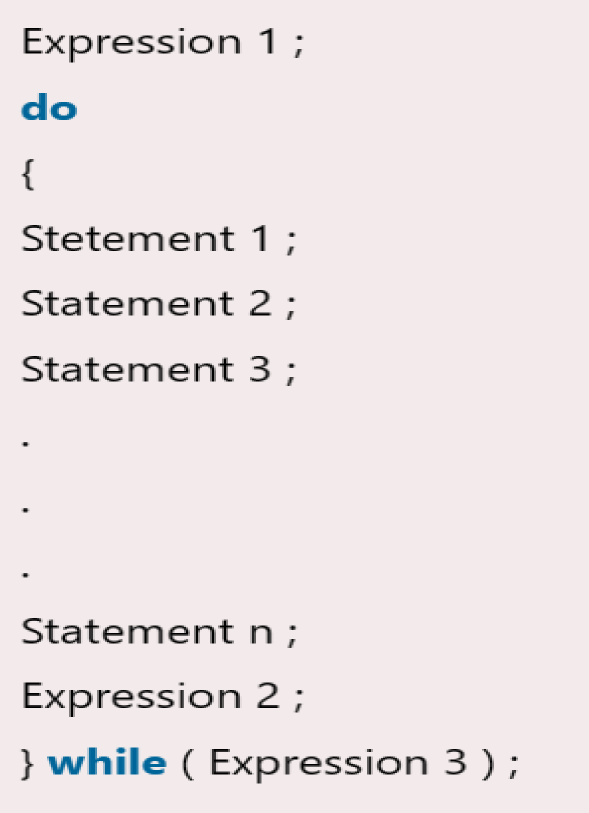
Output



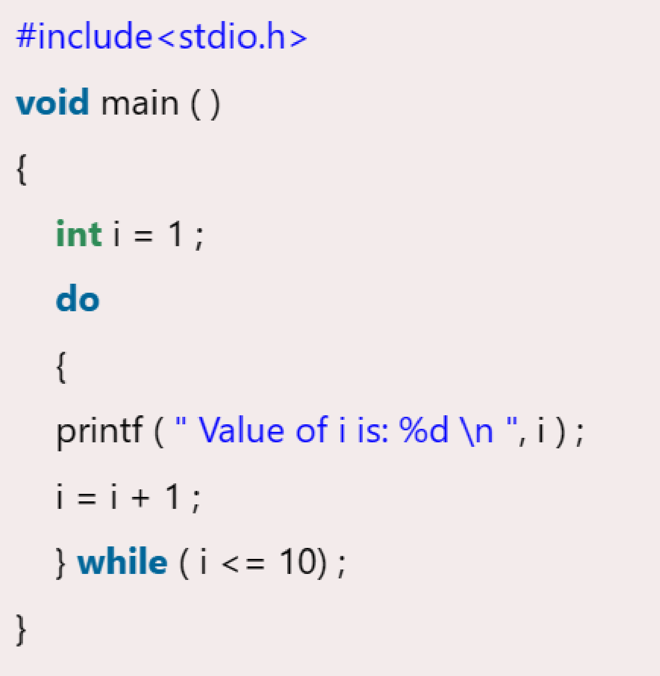
```
Value of i is: 1
Value of i is: 2
Value of i is: 3
Value of i is: 4
Value of i is: 5
Value of i is: 6
Value of i is: 7
Value of i is: 8
Value of i is: 9
Value of i is: 10
```

1. Using various loops in the program:

3. Using do-while loop:



```
Expression 1 ;
do
{
Stetement 1 ;
Statement 2 ;
Statement 3 ;
.
.
.
Statement n ;
Expression 2 ;
} while ( Expression 3 ) ;
```

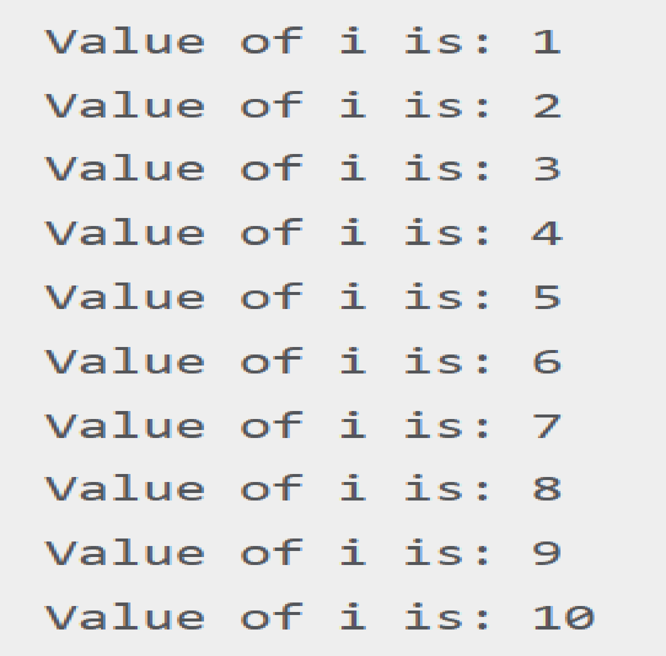


```
#include<stdio.h>
void main ( )
{
   int i = 1 ;
   do
   {
   printf ( " Value of i is: %d \n ", i ) ;
   i = i + 1 ;
   } while ( i <= 10) ;
}
```

1. Using various loops in the program:

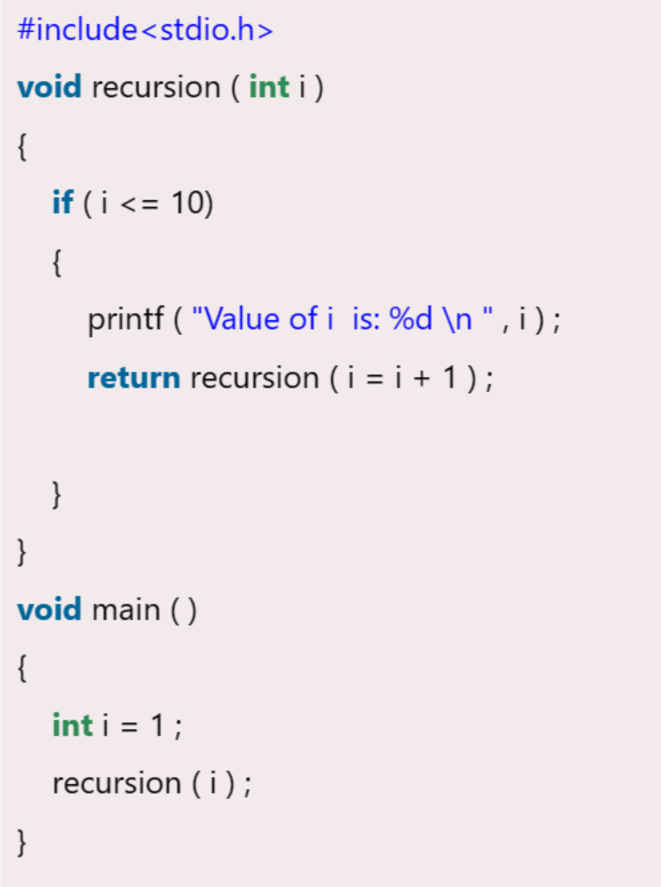
3. Using do-while loop:

Output



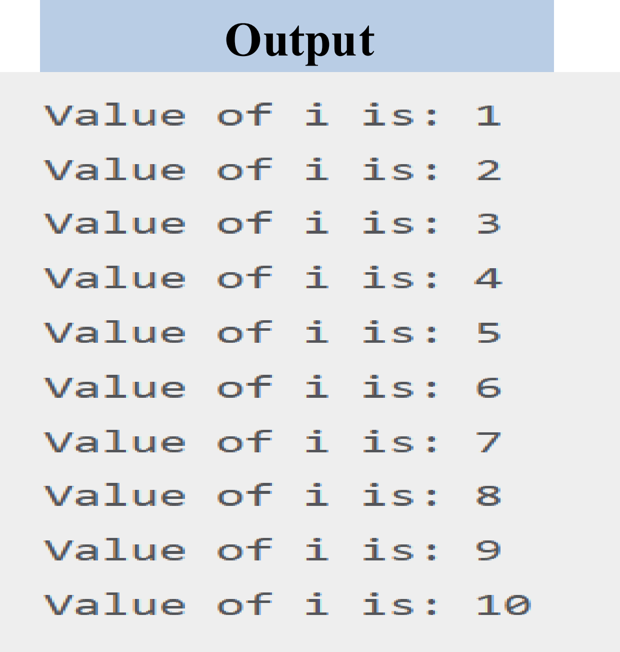
```
Value of i is: 1
Value of i is: 2
Value of i is: 3
Value of i is: 4
Value of i is: 5
Value of i is: 6
Value of i is: 7
Value of i is: 8
Value of i is: 9
Value of i is: 10
```

2. Using recursion in the program:



```
#include<stdio.h>
void recursion ( int i )
{
    if ( i <= 10)
    {
        printf ( "Value of i  is: %d \n " , i ) ;
        return recursion ( i = i + 1 ) ;

    }
}
void main ( )
{
    int i = 1 ;
    recursion ( i ) ;
}
```



Output

```
Value of i is: 1
Value of i is: 2
Value of i is: 3
Value of i is: 4
Value of i is: 5
Value of i is: 6
Value of i is: 7
Value of i is: 8
Value of i is: 9
Value of i is: 10
```

2. Using recursion in the program:

Definition

- Iterative and recurrence relations are mathematical expressions used to describe functions in terms of their own values at smaller inputs.
- They are commonly employed in algorithm analysis to determine the time complexity of recursive algorithms.

Base Case

- Both types of relations involve a base case, which defines the value of the function for the smallest input size.
- The base case serves as the termination condition for recursive calls.

Practice Questions

Q1: Write an iterative function to calculate the nth Fibonacci number. What is the 10th Fibonacci number?

Q2: Write an iterative function to calculate the factorial of a given number n. Calculate the factorial of 6 using this function.

Q3: Write an iterative function to calculate the sum of all elements in an array. Given the array '[4,5,6,7,8]' , what is the sum?

Q4: Write an iterative function to count the number of digits in a given number. How many digits are there in the number 789456?

Practice Questions (Answers)

A1:

```
def fibonacci(n):
    a, b = 0, 1
    for _ in range(n):
        a, b = b, a + b
    return a
```

A2:

```
def factorial(n):
    result = 1
    for i in range(1, n + 1):
        result *= i
    return result
```

Practice Questions (Answers)

A3:

```
def sum_array(arr):
    total = 0
    for num in arr:
        total += num
    return total
```

A4:

```
def sum_of_digits(n):
    total = 0
    while n != 0:
        total += n % 10
        n //= 10
    return total
```

Practice Questions- Try Yourself

Q1: Write an iterative function to reverse a given string. What is the reverse of the string "datastructure"?

Q2: While the Tower of Hanoi is typically solved using recursion, describe an iterative approach using stacks to simulate the process. How many moves are required to solve the Tower of Hanoi for 3 disks?

Q3: Write an iterative function to find the sum of the digits of a given number. What is the sum of the digits of the number 4321?

Q4: Write an iterative function to find the GCD of two numbers using Euclid's algorithm. What is the GCD of 56 and 98?

References

- https://archive.org/details/datastructuresal0000weis_j8p1
- https://edutechlearners.com/download/Introduction_to_algorithms-3rd%20Edition.pdf
- Classification-of-data-structure.png (890×477) (geekspartner.com)
- memory-allocation.png (559×397) (scientecheasy.com)
- https://www.sanfoundry.com/1000-data-structure-questions-answers/
- https://www.tutorialspoint.com/data_structures_algorithms/array_data_structure.htm
- https://media.geeksforgeeks.org/wp-content/uploads/20220816162225/Queue.png
- https://www.simplilearn.com/ice9/free_resources_article_thumb/Linked-List-Soni/representation-of-linked-list.png
- https://miro.medium.com/v2/resize:fit:7954/1*q0tQ1svUWfN05kWsZ_Uo2g.jpeg
- https://i2.wp.com/techvidvan.com/tutorials/wp-content/uploads/sites/2/2020/03/various-java-abstract-data-types.jpg?w=720&ssl=1

References

- Abstract Data Type (ADT) in Data Structure (prepbytes.com)
- https://interviewprep.org/abstract-data-type-interview-questions/
- IMO--Class-2-Maths-Olympiad-Question-Paper-1.jpg (998×1411) (management.ind.in)
- 96998994-193f1c00-1552-11eb-9dc4-395030b8a295.png (1272×624) (user-images.githubusercontent.com)
- u5myWkH-roLTh0iGBu-7Jbc_XWWbX9lv9o-VuHWxZnGiSpDS5vKZVQqkQaN2cSKxKA (512×250) (googleusercontent.com)
- 64360469.jpg (700×325) (toiimg.com)
- Bing Videos
- First+Hired+Last+Fired.jpg (1066×1600) (bp.blogspot.com)
- google-graph-768x432.jpg (768×432) (bizadmark.com)
- 1*r1ckz81ipEFmOWO8zEfUfw.png (1080×608) (medium.com)
- ds-introduction-768x411.png (768×411) (studywithgenius.in)
- stacks3-l.jpg (1024×768) (slideserve.com)
- Stack Abstract Data Type - Search Images (bing.com)

References

- Data-structures.png (685×497) (softwaretestinghelp.com)
- What Are Static and Dynamic Data Structure? - Scaler Topics
- array data structure - Search Images (bing.com)
- dc4c998cfc52b4c997701166efbe4b97.png (900×600) (pinimg.com)
- maxresdefault.jpg (1280×720) (ytimg.com)
- Introduction to Getting Started with Array Data Structure – GeeksforGeeks
- static.png (1000×500) (geeksforgeeks.org)
- java-two-dimensional-array.png (485×324) (scientecheasy.com)
- 3D-array.jpg (1107×551) (geeksforgeeks.org)
- time-complexity-in-data-structure-examples.jpeg (1250×1250) (thetapacademy.com)
- time-space-tradeoff-l.jpg (1024×768) (slideserve.com)
- Array Data Structure (tutorialspoint.com)

References

- (1250×1250) (thetapacademy.com)
- time-space-tradeoff-l.jpg (1024×768) (slideserve.com)
- Array Data Structure (tutorialspoint.com)
- http://exploreshaifali.github.io/2014/02/13/Space-Complexity/
- https://www.geeksforgeeks.org/a-time-complexity-question/
- https://www.interviewcake.com/article/java/big-o-notation-time-and-space-complexity
- https://discuss.codechef.com/t/finding-the-space-complexity-of-an-algorithm
- https://en.wikipedia.org/wiki/Time_complexity
- https://www.tutorialspoint.com/data_structures_algorithms/asymptotic_analysis.htm
- What is iteration in data structure - javatpoint

THANK YOU