CSA0358 DATA STRUCTURES WITH GRAPH ALGORITHMS

DAY-1:(08/08/2023)

QUESTION 1:

Write a C program to check the number is even or odd.

CODE:

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

OUTPUT:

```
enter a number:8
8 is an even number
--------------------------------
Process exited after 9.39 seconds with return value 0
Press any key to continue . . .
```

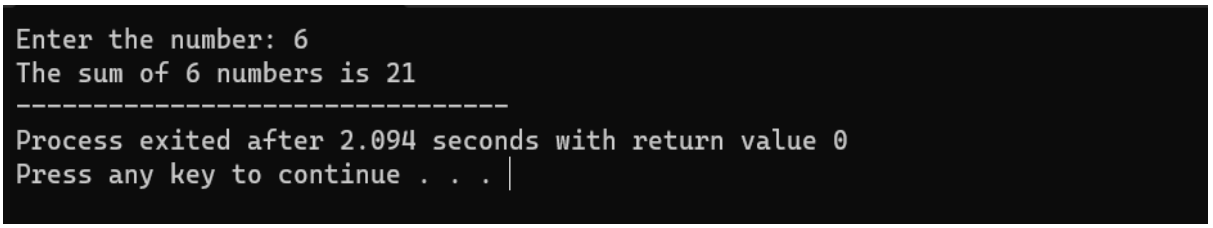
QUESTION 2:

Write a C program to find sum of first n numbers using for loop.

CODE:

```
#include<stdio.h>
int main(){
	int n,sum=0,i;
	printf("Enter the number: ");
	scanf("%d",&n);
	for(i=1;i<=n;i++){
		sum=sum+i;
	}
	printf("The sum of %d numbers is %d",n,sum);
	return 0;
}
```

OUTPUT:



QUESTION:3

Write a C program to find the sum of even numbers using while loop.

CODE:

```
#include<stdio.h>
int main(){
	int i,n,even_sum=0;
	printf("Enter the nth number: ");
	scanf("%d",&n);
	while(i<=n){
		if(i%2==0){
			even_sum+=i;
		}
		i++;
	}
	printf("The sum of even numbers is %d",even_sum);
	return 0;
}
```

OUTPUT:

```
Enter the nth number: 8
The sum of even numbers is 20
--------------------------------
Process exited after 7.656 seconds with return value 0
Press any key to continue . . .
```

QUESTION:4

Write a C program to reverse the numbers.

CODE:

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

OUTPUT:

```
Enter the number: 1234
Thr reversed number is 4321
--------------------------------
Process exited after 4.183 seconds with return value 0
Press any key to continue . . .
```

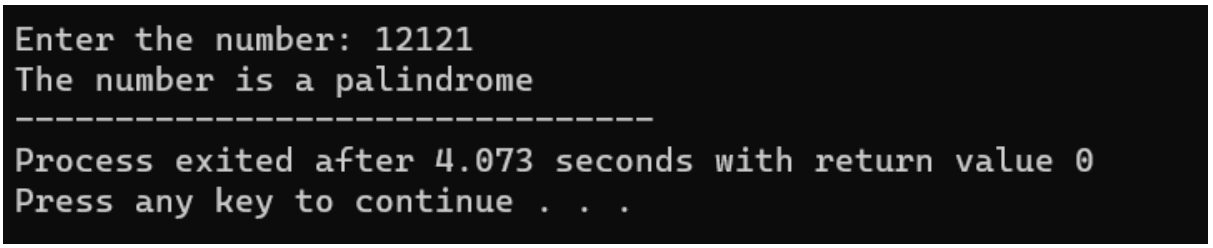
QUESTION:5

Write a C program to find the given number is a palindrome.

CODE:

```
#include<stdio.h>
int main(){
	int n,rem,rev,temp;
	printf("Enter the number: ");
	scanf("%d",&n);
	temp=n;
	while(n!=0){
		rem=n%10;
		rev=(rev*10)+rem;
		n=n/10;
	}
	if(rev==temp){
		printf("The number is a palindrome");
	}
	else{
		printf("The number is not a palindrome");
	}
	return 0;
}
```

OUTPUT:



QUESTION:6

Write a C program to find the given number is Armstrong or not.

CODE:

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

```
        n=n/10;
    }
    if(res==temp){
        printf("The number is an armstrong number");
    }
    else{
        printf("The number is not an armstrong number");
    }
    return 0;
}
```

OUTPUT:

```
Enter a number: 153
The number is an armstrong number
--------------------------------
Process exited after 10.99 seconds with return value 0
Press any key to continue . . .
```

QUESTION:7

Write a C program to find the factorial without recursion.

CODE:

```
#include<stdio.h>
int main(){
    int i,n,fact=1;
    printf("Enter the number: ");
    scanf("%d",&n);
    for(i=1;i<=n;i++){
        fact=fact*i;
    }
    printf("The factorial of %d is %d",n,fact);
}
```

OUTPUT:

```
Enter the number: 8
The factorial of 8 is 40320
--------------------------------
Process exited after 3.56 seconds with return value 0
Press any key to continue . . .
```

QUESTION:8

Write a C program to find the factorial with recursion.

CODE:

```
#include<stdio.h>

long factorial(int n)
{
  if (n == 0)
    return 1;
  else
    return(n * factorial(n-1));
}

int main()
{
  int number;
  long fact;
  printf("Enter a number: ");
  scanf("%d", &number);

  fact = factorial(number);
  printf("Factorial of %d is %ld\n", number, fact);
  return 0;
}
```

OUTPUT:

```
Enter a number: 6
Factorial of 6 is 720

--------------------------------
Process exited after 9.477 seconds with return value 0
Press any key to continue . . .
```

QUESTION:9

Write a C program to generate fibonacci series without recursion.

CODE:

```
#include<stdio.h>
int main(){
```

```
        int n,n1=0,n2=1,n3,i;
        printf("Enter the number of terms: ");
        scanf("%d",&n);
        printf("The fibonacci series is:\n%d\n%d ",n1,n2);
        for(i=3;i<=n;i++){
                n3=n1+n2;
                n1=n2;
                n2=n3;
                printf("\n%d",n3);
        }
        return 0;
}
```

OUTPUT:

```
Enter the number of terms: 8
The fibonacci series is:
0
1
1
2
3
5
8
13
--------------------------------
Process exited after 9.126 seconds with return value 0
Press any key to continue . . .
```

QUESTION:10

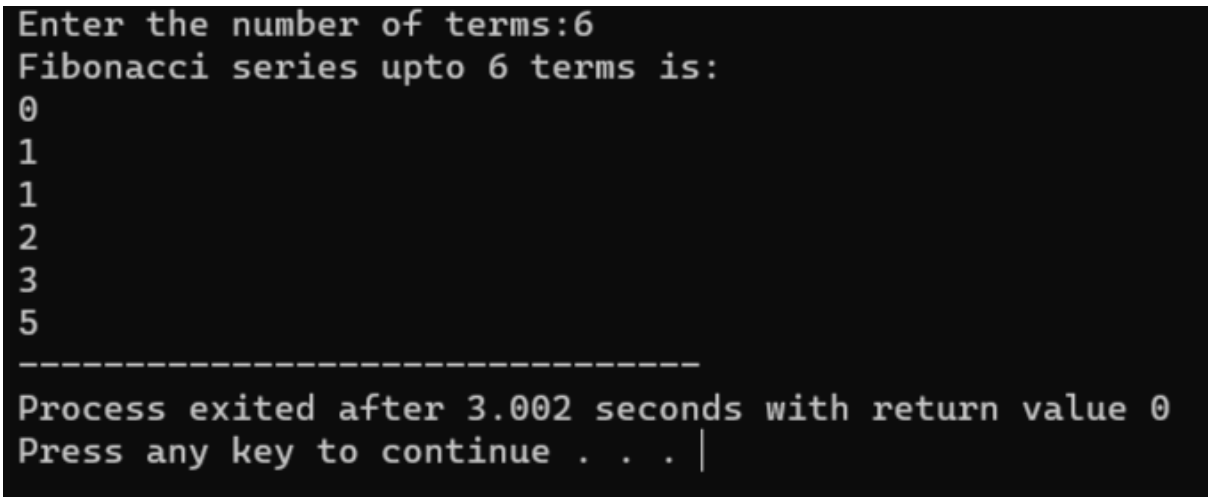
Write a C program to generate fibonacci series with recursion.

CODE:

```
#include<stdio.h>
int fibonacci(int n){
        if(n<=1){
                return n;
        }
        else{
                return fibonacci(n-1)+fibonacci(n-2);
        }
}
```

```
int main(){
        int n,i;
        printf("Enter the number of terms:");
        scanf("%d",&n);
        printf("Fibonacci series upto %d terms is:",n);
        for(i=0;i<n;i++){
                printf("\n%d",fibonacci(i));
        }
        return 0;
}
```

OUTPUT:



QUESTION:11

Write a C program to search a particular element in an array using Linear search.

CODE:

```
#include <stdio.h>
int linearSearch(int a[], int n, int val) {
  for (int i = 0; i < n; i++)
    {
       if (a[i] == val)
       return i+1;
    }
  return -1;
}
int main() {
  int n;
  int a[100];
```

```
  printf("Enter the size of the array: ");
  scanf("%d",&n);
  printf("Enter the elements of the array: ");
  for(int j=0;j<n;j++){
        scanf("%d",&a[j]);
  }
  int val;
  printf("Enter the value to be searched: ");
  scanf("%d",&val);
  int res = linearSearch(a, n, val);
  printf("The elements of the array are:\n");
  for (int i = 0; i < n; i++)
  printf("%d ", a[i]);
  printf("\nElement to be searched is : %d", val);
  if (res == -1)
  printf("\nElement is not present in the array");
  else
  printf("\nElement is present at %d position of array", res);
  return 0;
}
```

OUTPUT:

```
Enter the size of the array: 5
Enter the elements of the array: 2
4
6
8
10
Enter the value to be searched: 8
The elements of the array are:
2 4 6 8 10
Element to be searched is : 8
Element is present at 4 position of array
--------------------------------
Process exited after 29.21 seconds with return value 0
Press any key to continue . . .
```

QUESTION:12

Write a C program to search a particular element in an array using Binary search.

CODE:

```
#include <stdio.h>
int iterativeBinarySearch(int array[], int start_index, int end_index, int
element){
  while (start_index <= end_index){
    int middle = start_index + (end_index- start_index )/2;
    if (array[middle] == element)
      return middle;
    if (array[middle] < element)
      start_index = middle + 1;
    else
      end_index = middle - 1;
  }
  return -1;
}
int main(void){
  int array[] = {1,4,7,9,16,56,70};
  int n = 7;
  int element = 16;
  int found_index = iterativeBinarySearch(array, 0, n-1, element);
  if(found_index == -1 ) {
    printf("Element not found in the array ");
  }
  else {
    printf("Element found at index : %d",found_index);
  }
  return 0;
}
```

OUTPUT: