

Control Structure

26-10-25

Aim – Implement a program to demonstrate control structure, write code to check whether it is Armstrong, generate Fibonacci series, LCM, GCD and grading system.

Theory –

If - Executes a block of code only when a specified condition evaluates to true, enabling decision-making in programs.

Else if - Checks multiple conditions sequentially; executes the first true condition's block, providing more flexible decision-making than simple if.

While - Repeats a block of code as long as a specified condition remains true, enabling controlled iteration.

Break - Immediately exits the nearest enclosing loop or switch, stopping further iterations or case execution prematurely.

A1.

```
#include<stdio.h>

void main()
{
    int n,count=0,result=0,mul=1,cnt,rem;
    printf("Armstrong Numbers\n");
    printf("enter a number:");
    scanf("%d",&n);
    int q=n;
    while(q!=0){
        q=q/10;
        count++;
    }
    cnt=count;
    q=n;
    while(q!=0){
        rem=q%10;
        while(cnt!=0){
            mul= mul*rem;
            cnt--;
        }
        result = result+mul;
    }
}
```

```

cnt = count;
q = q/10;
mul = 1;
}
if(result==n){
printf("%d is an armstrong number",n);
} else {
printf("%d is an not armstrong number",n);
}
}

```

```

Armstrong Numbers
enter a number:5
5 is an armstrong number
Process returned 24 (0x18)    execution time : 1.561 s
Press any key to continue.
|
```

A2.

```

#include <stdio.h>
void main(){
int a=0,b=1,c,i,n;
printf("Enter a number:");
scanf("%d",&n);
printf("Fibonacci series \n0\n1\n");
for(i=1; i<=n-2;i++){
c=a+b;
printf("%d\n",c);
a=b;
b=c;
}
}
```

```
Enter a number:5
Fibonacci series
0
1
1
2
3

Process returned 4 (0x4)    execution time : 0.808 s
Press any key to continue.
```

A3.

```
#include <stdio.h>

void main()
{
    int num1, num2, lcm;
    printf("LCM\n");
    printf("Enter first number: ");
    scanf("%d", &num1);
    printf("Enter second number: ");
    scanf("%d", &num2);
    lcm = (num1 > num2) ? num1 : num2;
    while (1) {
        if (lcm % num1 == 0 && lcm % num2 == 0){
            printf("The LCM of %d and %d is %d.\n", num1, num2, lcm);
            break;
        }
        lcm++;
    }
}
```

```
LCM
Enter first number: 20
Enter second number: 65
The LCM of 20 and 65 is 260.
```

```
Process returned 29 (0x1D)    execution time : 7.049 s
Press any key to continue.
```

A4.

```
#include <stdio.h>

void main()
{
    int n1,n2,gcd;
    printf("GCD\n");
    printf("Enter first number:");
    scanf("%d", &n1);
    printf("Enter second number:");
    scanf("%d", &n2);
    if(n1<n2){
        gcd=n1;
    } else {
        gcd=n2;
    }
    while(n1% gcd!=0 || n2% gcd!=0){
        gcd--;
    }
    if(gcd==1){
        printf("GCD doesn't exist");
    } else {
        printf("GCD= %d",gcd);
    }
}
```

```
GCD
Enter first number:50
Enter second number:225
GCD= 25
Process returned 7 (0x7)    execution time : 6.118 s
Press any key to continue.
|
```

A5.

```
#include<stdio.h>

void main()
{
    int marks;
    printf("Enter your marks (0 - 100): ");
    scanf("%d", &marks);
    if (marks >= 90 && marks <= 100)
        printf("Grade: A\n");
    else if (marks >= 80)
        printf("Grade: B\n");
    else if (marks >= 70)
        printf("Grade: C\n");
    else if (marks >= 60)
        printf("Grade: D\n");
    else if (marks >= 50)
        printf("Grade: E\n");
    else if (marks >= 0)
        printf("Grade: F (Fail)\n");
    else
        printf("Invalid marks entered.\n");
}
```

```
Enter your marks (0 - 100): 94
```

```
Grade: A
```

```
Process returned 0 (0x0)    execution time : 4.283 s
```

```
Press any key to continue.
```

Conclusion

Through this assignment, we explored various control structures like if, else if, while, and break to solve real-world problems. By implementing programs for Armstrong numbers, Fibonacci series, LCM, GCD, and grading, we learned how decision-making and iteration form the backbone of logical programming, enhancing problem-solving skills effectively.