

AMRITA SCHOOL OF ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

19CSE205 - Program Reasoning

LAB ASSIGNMENT-6

Class: 3rd Semester CSE-E

Course Faculty: Dr.S.Padmavathi

DATE: 15-9-2020

HW Assignment CO-2

MARKS:20

CO1	Understand the necessity of proving correctness of programs with respect to formal specification or property
CO2	Apply program verification techniques to prove and analyze correctness of programs.
CO3	Understand the problems associated with concurrent programs and their effects on their behavioural correctness
CO4	Understand and use few state-of-the-art tools to model and verify sequential and concurrent systems.

PART-A (each 5 marks).

Identify the output condition and invariant.

Derive and verify Intra loop and exit loop conditions.

Write the suitable conditions for Total Correctness.

Verify the code in Frama-C

1. By Arithmetic Progression, we know, for any series, the sum of numbers is given by;

$$S_n = 1/2 \times n [2a + (n-1)d] \dots\dots\dots(2)$$

Where,

n = number of digits in the series

a = First term of an A.P

d= Common difference in an A.P

The following code adds sum of n elements of a sequence= 5,9,13,... Identify the loop invariant. Verify the intra loop and exit loop condition in Frama-c.

```
{ i = 5;  
    s = 5;
```

```

F=4*(n-1)
    @invariant ...
    while (i != f) {
        i = i + 4;
        s = s + i;
    }
}

```

2. Program to find GCD:

```

int x, y, temp;
int g = 1;
while(y != 0){
    x = x % y;
    temp = y;
    y = x;
    x = temp;
    g = x;
}
return g;

```

3. TO REVERSE A NUMBER

```

int n, rev = 0, remainder;
while (n != 0) {
    remainder = n % 10;
    rev = rev * 10 + remainder;
    n /= 10;
}
Return rev

```

4. Sum of digits

```

int n, s = 0, remainder;
while (n != 0) {
    remainder = n % 10;
    s = s + remainder;
    n /= 10;
}
Return s

```