# PROCESS INSTRUMENTATION

**Name: Vivek Chandrashekhar Rugale**

**GR No.: 11810369**

**Div.: TY-C**

**Roll No.: 24 (Batch B1)**

## LAB 6:

### Aim – Study of flow totalizer and its implementation

Implementation of flow totalizer logic in Python:

For constant flow-

In [4]:
```
t = int(input("Enter the time after which flow is to be calculated: "))
print("\nInput(mA)\tOutput(LPM)\tTotal Flow(L)\n")
for i in range(4,21):
    lpm = (6.25*i)-25
    TotalFlow = lpm*t
    print(str(i) + "\t\t" + str(lpm) + "\t\t" + str(TotalFlow))
    i = i+2
```

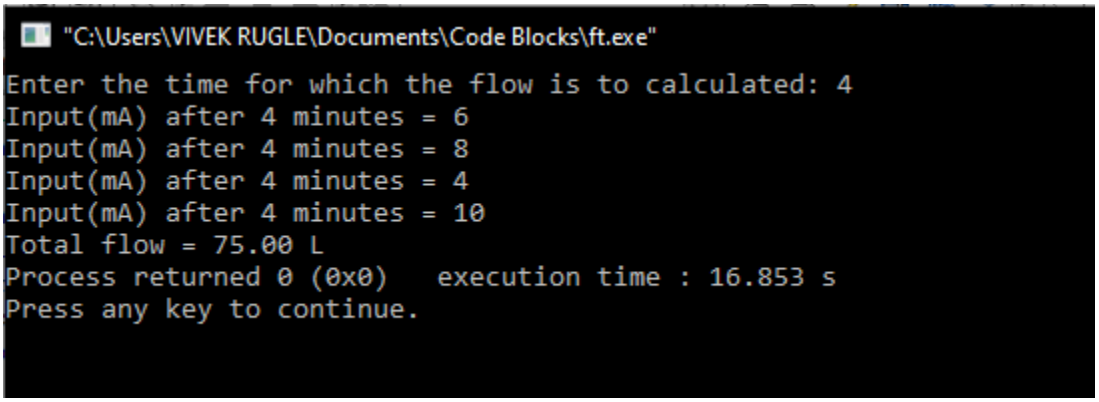
```
Enter the time after which flow is to be calculated: 3

Input(mA)       Output(LPM)     Total Flow(L)

4               0.0             0.0
5               6.25            18.75
6               12.5            37.5
7               18.75           56.25
8               25.0            75.0
9               31.25           93.75
10              37.5            112.5
11              43.75           131.25
12              50.0            150.0
13              56.25           168.75
14              62.5            187.5
15              68.75           206.25
16              75.0            225.0
17              81.25           243.75
18              87.5            262.5
19              93.75           281.25
20              100.0           300.0
```

For variable flow-

```c
#include<stdio.h>
int main()
{
int t,i;
float input,flow,total;
printf("Enter the time for which the flow is to calculated: ");
scanf("%d",&t);
for(i=1;i<=t;i++){
    printf("Input(mA) after %d minutes = ",t);
    scanf("%f",&input);
    flow = (6.25*input)-25;
    total = total + flow;
}
printf("Total flow = %.2f L",total);
return 0;
}
```




**Conclusion:**

Studied the flow totalizer logic and implemented it using Python. For variable flow, flow is calculated at each unit time and then added.