




# **INDIAN INSTITUTE OF TECHNOLOGY GOA**

**Name:** Aditya Rajesh Bawangade

**Roll Number:** 2103111

### Output:



```
PS D:\LECS AND MATERIAL\SEMESTER 4\Algorithm Design\Assignment4> ./a.exe
Enter the three variables namely x, y, z: 100
1
2
5 17
2 26
2 26
2 28
PS D:\LECS AND MATERIAL\SEMESTER 4\Algorithm Design\Assignment4>
```

Note that all numbers are in **base 37**.

Analysis of the time complexity:

The constructor takes  **$O(\log_b(\text{Val}))$**  times, as it converts the number to base b format and stores it in the private member vector.

The operator overloading of the + function takes  **$O(\max(\text{vector1.size}, \text{vector2.size}))$**  as the while loop operates that long and the zeroes are popped after the loop stops.

The operator overloading of the subtraction operator takes the same amount of time as the first + operator i.e.,  **$O(\max(\text{vector1.size}, \text{vector2.size}))$** .

The multiplication operator takes  **$O(\text{vector1.size} * \text{vector2.size})$**  time.

The division operator takes  **$O((\text{vector1.size} - \text{vector2.size}) * \text{vector2.size})$**  time. Since, It is done using the base b division method of dividing the problem into  $1/b$  of its original size and then adding the popped bit again after left shifting. Then, we loop until the remainder is smaller than the denominator by subtracting the denominator from it and incrementing the quotient by one each time.

The ++ operator takes the same amount of time as the constructor i.e.,  **$O(\log_b(\text{Val}))$**  time.