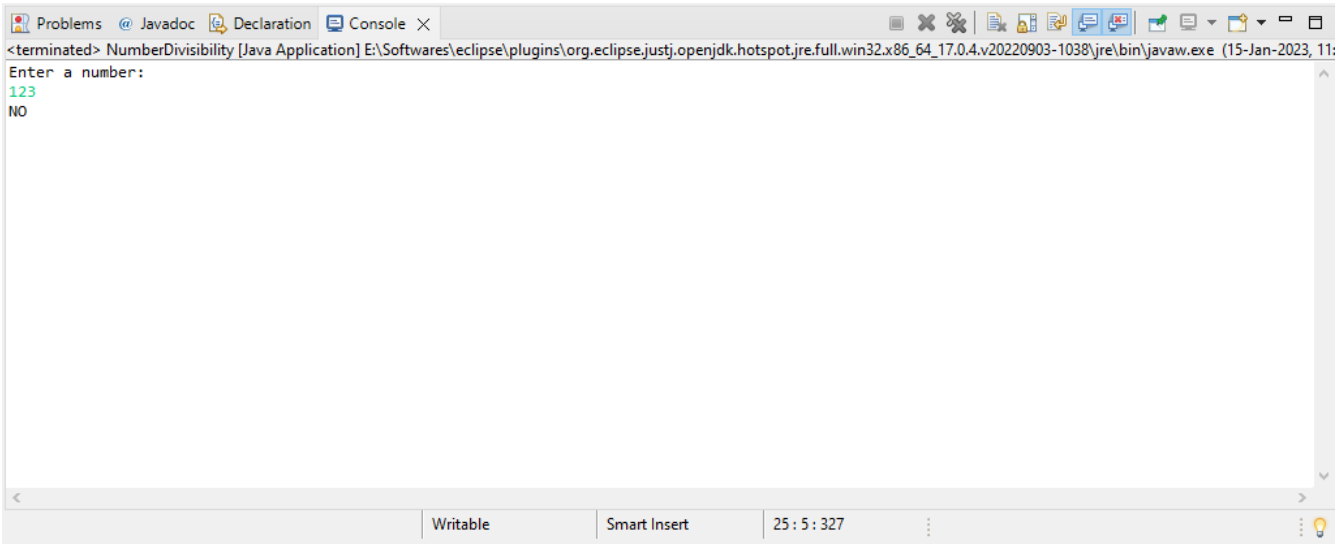
1. Write a program to check whether a number given as input is divisible by the sum of its digits.

```java
package com.numberDivisibility.info;

import java.util.Scanner;

class NumberDivisibility
{

   static String isDivisible(long n)
   {
         long temp = n;

         int sum = 0;
         while (n != 0)
         {
               int k = (int) n % 10;
               sum += k;
               n /= 10;
         }

         if (temp % sum == 0)
               return "YES";

         return "NO";
   }

   public static void main(String []args)
   {
         System.out.println("Enter a number: ");
         Scanner sc = new Scanner(System.in);
         int n = sc.nextInt();
         System.out.println(isDivisible(n));
         sc.close();
   }
}
```

Name: Bhoite Sanket Vikas PRN: 220960920015

Output:



2. Write a C program to create three threads which access the thread function which displays the string data. Allow only one thread to access the thread function at a time.

```
#include <stdio.h>
#include <pthread.h>

pthread_mutex_t mutex;

void *thread_function(void *arg)
{
    // Acquire a lock before printing data
    pthread_mutex_lock(&mutex);

    printf("Thread Function: %s\n", (char *)arg);

    // Release the lock
    pthread_mutex_unlock(&mutex);
```

```
    return NULL;
}

int main()
{
    pthread_t thread1, thread2, thread3;
    char *msg1 = "Thread 1";
    char *msg2 = "Thread 2";
    char *msg3 = "Thread 3";

    // Initialize the mutex
    pthread_mutex_init(&mutex, NULL);

    // Create the threads
    pthread_create(&thread1, NULL, thread_function, (void *)msg1);
    pthread_create(&thread2, NULL, thread_function, (void *)msg2);
    pthread_create(&thread3, NULL, thread_function, (void
    *)msg3);

    // Join the threads
    pthread_join(thread1, NULL);
    pthread_join(thread2, NULL);
    pthread_join(thread3, NULL);

    // Destroy the mutex
    pthread_mutex_destroy(&mutex);

    return 0;
}
```

Name: Bhoite Sanket Vikas PRN: 220960920015

Output: