DAA WEEK2 SUBMISSION

Name: Davasam Karthikeya   Section: AIMLB   Reg No: 230962326

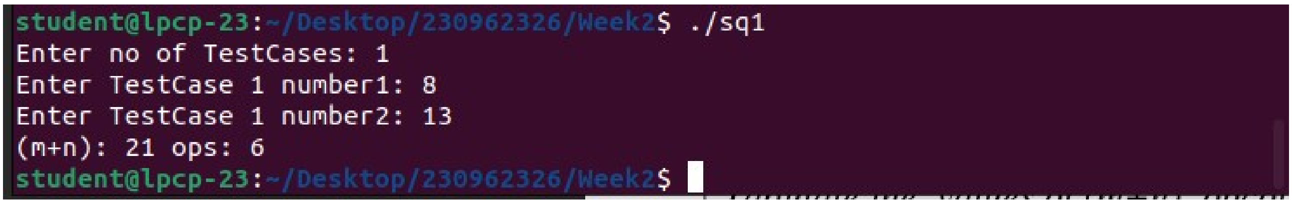
Date: 11/01/2025

1) write a program to implement EUCLID’s algorithm to find GCD of two numbers

Code:

```
#include <stdio.h>
int EuclidGCD(unsigned int m, unsigned int n) {
    int r,opcount = 0;
    while(n!=0) {
        opcount++;
        r = m %n;
        m = n;
        n = r;
    }
    return opcount;
}

int main(){
    int TestCases;
    printf("Enter no of TestCases: ");scanf("%d",&TestCases);
    int nums1[TestCases],nums2[TestCases];
    for(int i = 0; i < TestCases; i++){
        printf("Enter TestCase %d number1: ",i+1);scanf("%d",&nums1[i]);
        printf("Enter TestCase %d number2: ",i+1);scanf("%d",&nums2[i]);
    }
    for(int i = 0; i < TestCases; i++){
        int ops = EuclidGCD(nums1[i],nums2[i]);
        printf("(m+n): %d ops: %d \n",nums1[i]+nums2[i],ops);
    }
}
```

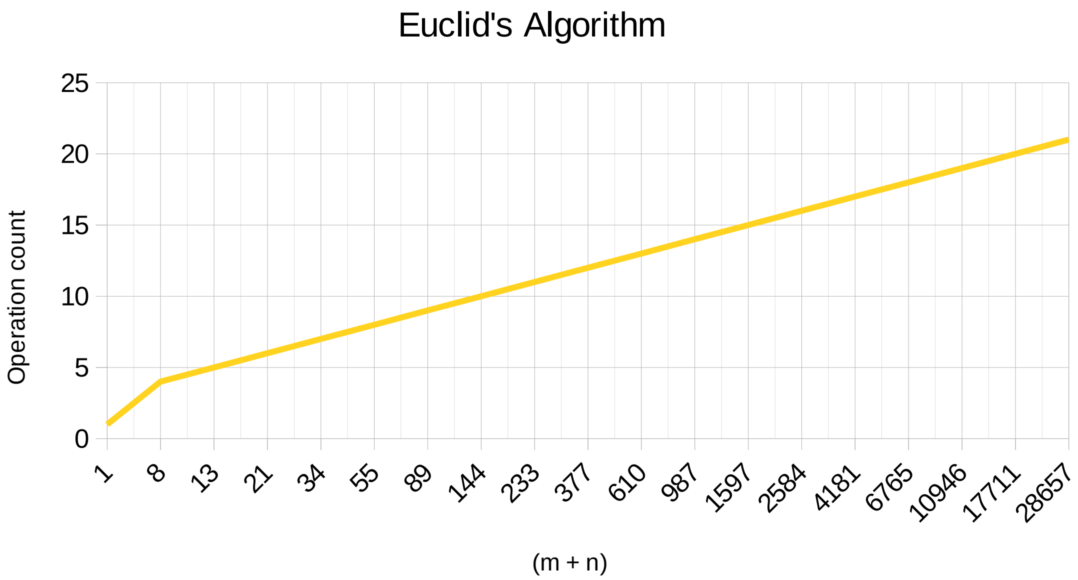
Sample Input/Output:-



```
student@lpcp-23:~/Desktop/230962326/Week2$ ./sq1
Enter no of TestCases: 1
Enter TestCase 1 number1: 8
Enter TestCase 1 number2: 13
(m+n): 21 ops: 6
student@lpcp-23:~/Desktop/230962326/Week2$ 
```

Graph:

}

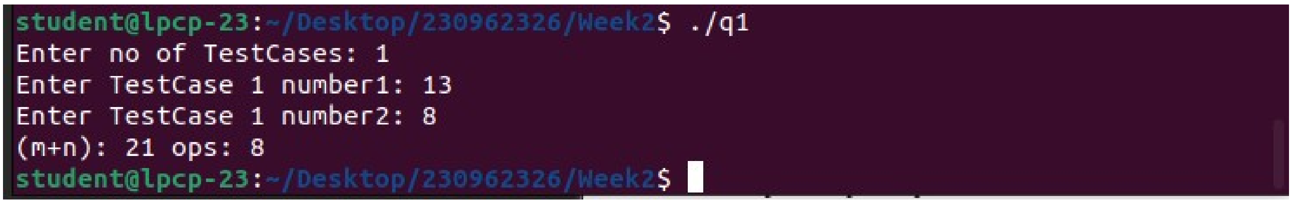


2) Write a program to find GCD using consecutive integer checking method and analyze its time efficiency.

Code:-

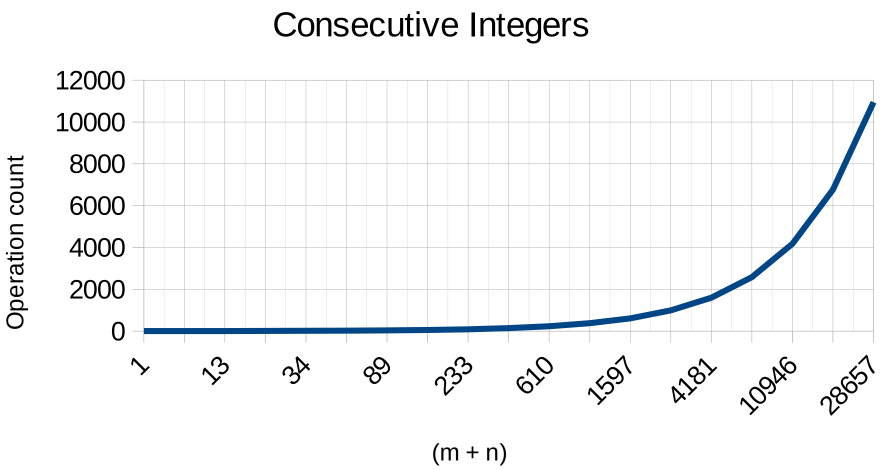
```
#include <stdio.h>

int min(int number1, int number2){
    return (number1<number2)?number1:number2;
}

int GCD(int number1, int number2){
    int opercount = 0;
    int nmin = min(number1,number2);
    for(;nmin > 0;nmin--){
        opercount++;
        if((number1 % nmin == 0) && (number2 % nmin == 0))return opercount;
    }
}

int main(){
    int TestCases;
    printf("Enter no of TestCases: ");scanf("%d",&TestCases);
    int nums1[TestCases],nums2[TestCases];
    for(int i = 0; i < TestCases; i++){
        printf("Enter TestCase %d number1: ",i+1);scanf("%d",&nums1[i]);
        printf("Enter TestCase %d number2: ",i+1);scanf("%d",&nums2[i]);
    }
    for(int i = 0; i < TestCases; i++){
        int ops = GCD(nums1[i],nums2[i]);
        printf("(m+n): %d ops: %d \n",nums1[i]+nums2[i],ops);
    }
}
```

Sample Output/Input:



```
student@lpcp-23:~/Desktop/230962326/Week2$ ./q1
Enter no of TestCases: 1
Enter TestCase 1 number1: 13
Enter TestCase 1 number2: 8
(m+n): 21 ops: 8
student@lpcp-23:~/Desktop/230962326/Week2$ 
```

Graph:

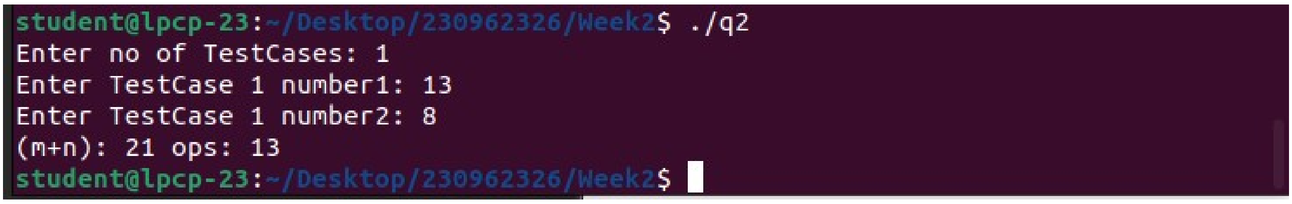


3) Write a program to find GCD using middle school method and analyze its time efficiency.

Code:

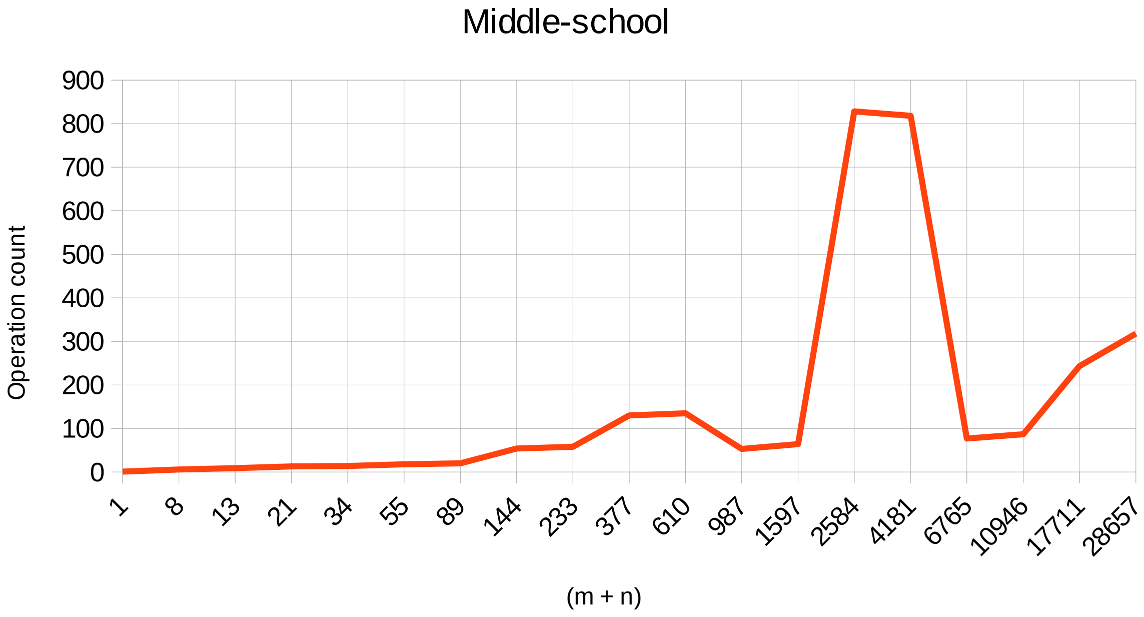
```
#include <stdio.h>
#include <stdlib.h>
int PrimeFactors(int *factors, int number){
    int k = 0,count = 0;
    while(number %2 == 0){
        count++;
        factors[k++] = 2;
        number /= 2;
    }
    for(int i = 3;i<= number;i+=2){
        count++;
        while(number % i == 0){
            count++;
            factors[k++] = i;
            number /= i;
        }
    }
    return count;
}

int MiddleSchoolGCD(int number1, int number2){
    //int factors1[100],factors2[100];
    int *factors1 = (int*)calloc(100,sizeof(int));
    int *factors2 = (int*)calloc(100,sizeof(int));
    int gcd = 1;
    int ops1 = PrimeFactors(&factors1[0],number1);
    int ops2 = PrimeFactors(&factors2[0],number2);
    int i = 0, j = 0;
    while(factors1[i] && factors2[j]){
        if(factors1[i] < factors2[j])i++;
        else if(factors1[i] > factors2[j])j++;
        else {i++;j++;gcd*=factors1[i];}
        ops1++;
    }
    return ops1 + ops2;
}

int main(){
    int TestCases;
    printf("Enter no of TestCases: ");scanf("%d",&TestCases);
    int nums1[TestCases],nums2[TestCases];
    for(int i = 0; i < TestCases; i++){
        printf("Enter TestCase %d number1: ",i+1);scanf("%d",&nums1[i]);
        printf("Enter TestCase %d number2: ",i+1);scanf("%d",&nums2[i]);
    }
    for(int i = 0; i < TestCases; i++){
        int ops = MiddleSchoolGCD(nums1[i],nums2[i]);
        printf("(m+n): %d ops: %d \n",nums1[i]+nums2[i],ops);
    }
}
```

Sample Input/Output:



```
student@lpcp-23:~/Desktop/230962326/Week2$ ./q2
Enter no of TestCases: 1
Enter TestCase 1 number1: 13
Enter TestCase 1 number2: 8
(m+n): 21 ops: 13
student@lpcp-23:~/Desktop/230962326/Week2$
```

Graph:


Middle-school
Operation count
(m + n)