Name: Brijesh Rameshbhai Rohit
Admission number: U19CS009

DISTRIBUTED SYSTEMS
ASSIGNMENT - 07

Simulate RPC (Create any one procedure on remote machine and call it from local machine)

Code=>

```
struct input
{
    int choice;
    char str1[50];
    char str2[50];
    int num1;
    int num2;
    int year;
    int arr[100];
};

struct output
{
    int flag;
    int gcd;
    float sqrt;
    int num1;
    int num2;
    int max;
    int min;
    float avg;
    char str[100];
    int arr[100];
};

program EXEC_PROG{
    version EXEC_VERS{
        output exec(input)=1;
    }=1;
}=0x2f2f2f2f;
```

CLIENT SIDE CODE =>

```
/*
* This is sample code generated by rpcgen.
* These are only templates and you can use them
* as a guideline for developing your own functions.
*/

#include "birju.h"


void
exec_prog_1(char *host, int ch)
{
   CLIENT *clnt;
   output  *result_1;
   input  exec_1_arg;

#ifndef DEBUG
   clnt = clnt_create (host, EXEC_PROG, EXEC_VERS, "udp");
   if (clnt == NULL) {
       clnt_pcreateerror (host);
       exit (1);
   }
#endif  /* DEBUG */

   exec_1_arg.choice=ch;

   switch (ch)
   {
   case 1:
       printf("Please enter the string : ");
       scanf("%s", exec_1_arg.str1);
       break;

   case 2:
       printf("Please enter a year : ");
       scanf("%d", &exec_1_arg.year);
       break;
```

```
   case 3:
       printf("Please enter first integer  : ");
       scanf("%d", &exec_1_arg.num1);
       printf("Please enter second integer : ");
       scanf("%d", &exec_1_arg.num2);
       break;

   case 4:
       printf("Please enter a number : ");
       scanf("%d", &exec_1_arg.num1);
       break;

   case 5:
       printf("Please enter first number  : ");
       scanf("%d", &exec_1_arg.num1);
       printf("Please enter second number : ");
       scanf("%d", &exec_1_arg.num2);
       printf("Values before swap are : %d %d\n", exec_1_arg.num1,
exec_1_arg.num2);
       break;

   case 6:
       printf("Please enter number of elements : ");
       scanf("%d", &exec_1_arg.year);
       printf("Please enter %d elements : ", exec_1_arg.year);
       for(int i = 0 ; i < exec_1_arg.year && i < 20; i++)
       {
           scanf("%d", &exec_1_arg.arr[i]);
       }
       break;
   case 7:
       printf("Please enter first string  : ");
       scanf("%s", exec_1_arg.str1);
       printf("Please enter second string : ");
       scanf("%s", exec_1_arg.str2);
       break;

   case 8:
       printf("Please enter first string  : ");
```

```
        scanf("%s", exec_1_arg.str1);
        printf("Please enter second string : ");
        scanf("%s", exec_1_arg.str2);
        break;

    case 9:
        printf("Please enter first string  : ");
        scanf("%s", exec_1_arg.str1);
        printf("Please enter second string : ");
        scanf("%s", exec_1_arg.str2);
        break;

    case 10:
        printf("Please enter number of elements : ");
        scanf("%d", &exec_1_arg.year);
        printf("Please enter %d elements : ", exec_1_arg.year);
        for(int i = 0 ; i < exec_1_arg.year ; i++)
        {
            scanf("%d", &exec_1_arg.arr[i]);
        }
        break;

    default:
        printf("Please enter a valid choice!!\n");
        return;
        break;
    }

    result_1 = exec_1(&exec_1_arg, clnt);
    if (result_1 == (output *) NULL) {
        clnt_perror (clnt, "call failed");
    }

    switch (ch)
    {
    case 1:
        if(result_1->flag)
            printf("%s is Palindrome!!\n", exec_1_arg.str1);
        else
```

```
            printf("%s is not a Palindrome!!\n", exec_1_arg.str1);
        break;
    case 2:
        if(result_1->flag)
            printf("%d is a LEAP!!\n", exec_1_arg.year);
        else
            printf("%d is not LEAP!!\n", exec_1_arg.year);
        break;
    case 3:
        printf("GCD(%d,%d) : %d\n", exec_1_arg.num1, exec_1_arg.num2,
result_1->gcd);
        break;
    case 4:
        printf("Square root of %d : %f\n",
exec_1_arg.num1,result_1->sqrt);
        break;
    case 5:
        printf("Values after swap are : %d %d\n", result_1->num1,
result_1->num2);
        break;
    case 6:
        printf("Maximum value in array : %d", result_1->max);
        printf("\nMinimum Value in array : %d", result_1->min);
        printf("\nAverage of all the values : %f\n", result_1->avg);
        break;
    case 7:
        if(result_1->flag > 0)
            printf("'%s' is greater than '%s'!!\n", exec_1_arg.str1,
exec_1_arg.str2);
        else if(result_1->flag == 0)
            printf("Both strings are same!!\n");
        else
            printf("'%s' is greater than '%s'!!\n", exec_1_arg.str2,
exec_1_arg.str1);
        break;
    case 8:
        if(result_1->flag)
            printf("%s string is substring of %s!!\n", exec_1_arg.str2,
exec_1_arg.str1);
```

```
        else
            printf("%s string is not substring of %s!!\n",
exec_1_arg.str2, exec_1_arg.str1);
        break;
    case 9:
        printf("Concatenated string : %s\n", result_1->str);
        break;
    case 10:
        printf("Reverse of array : ");
        for(int i = 0 ; i < exec_1_arg.year ; i++)
            printf("%d ", result_1->arr[i]);
        printf("\n");
        break;
    default:
        printf("Bye Bye!!\n");
        return;
        break;
    }

#ifndef DEBUG
    clnt_destroy (clnt);
#endif    /* DEBUG */
}


int
main (int argc, char *argv[])
{
    char *host;

    if (argc < 2) {
        printf ("usage: %s server_host\n", argv[0]);
        exit (1);
    }
    host = argv[1];
    int ch;
    char op='y';
    while (op=='y')
    {
```

```
        system("clear");
        printf("1  : To check if a string is palindrome.\n");
        printf("2  : Find if a year is leap.\n");
        printf("3  : find gcd of two number.\n");
        printf("4  : find square root of a given number.\n");
        printf("5  : Swap two variables without third variable.\n");
        printf("6  : Calculate maximum, mminimum and average.\n");
        printf("7  : Compare two strings.\n");
        printf("8  : Find whether a second string is substring.\n");
        printf("9  : Concatenate two strings.\n");
        printf("10 : Reverse elements of array.\n");
        printf("Please enter your choice : ");
        scanf("%d", &ch);
        exec_prog_1 (host,ch);
        printf("Do you want to continue? (y/n) : ");
        scanf("%s", &op);
    }
    printf("Hope you liked our service!!\n");
exit (0);
}
```

SERVER SIDE CODE=>

```c
/*
* This is sample code generated by rpcgen.
* These are only templates and you can use them
* as a guideline for developing your own functions.
*/

#include "birju.h"
#include <string.h>

int isPalindrome(char str[])
{
   int l=0;
   int h = strlen(str)-1;
   while (h>l)
   {
       if (str[l++] != str[h--])
           return 0;
   }
   return 1;
}

int isLeap(int year)
{
   if(year%400==0 || (year%100!=0 && year%4 == 0))
       return 1;
   return 0;
}

int gcd1(int a, int b)
{
   if(a==0)
       return b;
   return gcd1(b%a,a);
}

int isSubstr(char *str1,char *str2)
{
```

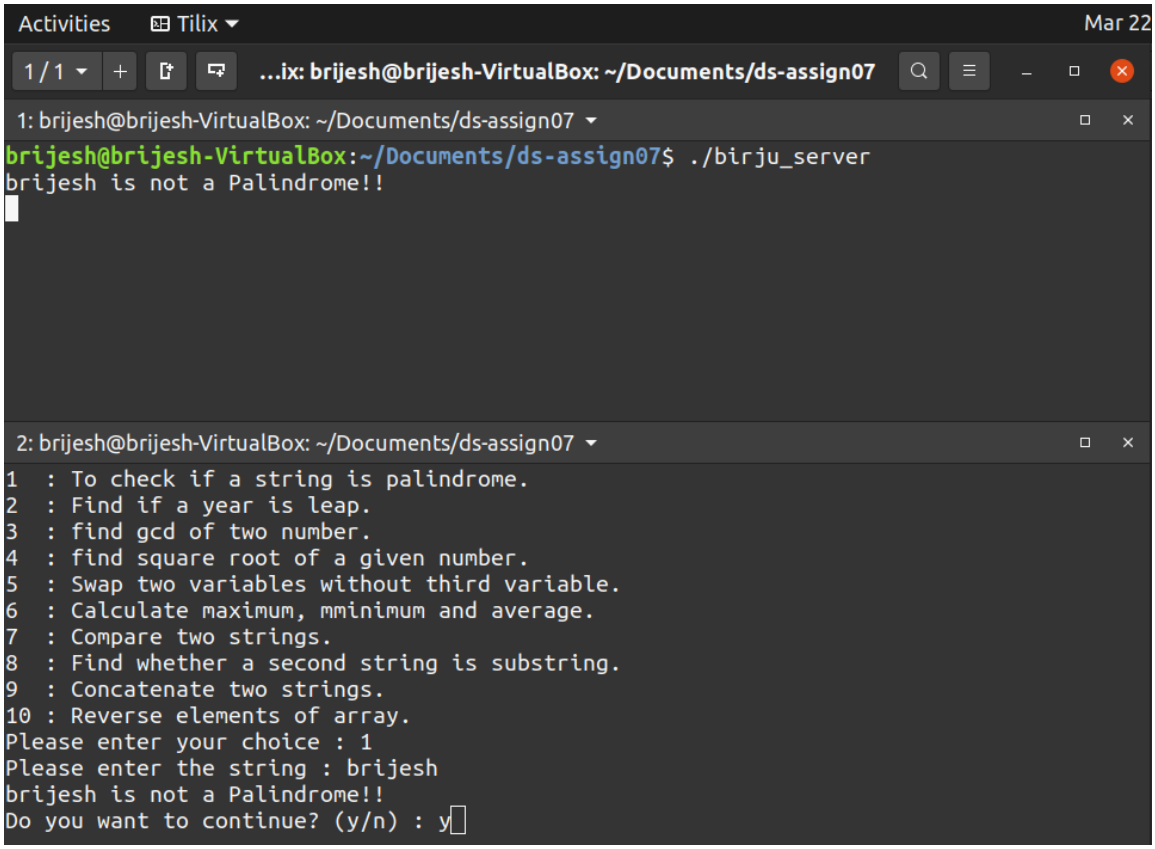
```
   int i=0,j;
   int n1=strlen(str1);
   int n2=strlen(str2);
   int flag;
   for (i = 0; i <= n1 - n2; i++)
   {
       for (j = i; j < i + n2; j++)
       {
           flag = 1;
           if (str1[j] != str2[j - i])
           {
               flag = 0;
               break;
           }
       }
       if (flag == 1)
           break;
   }
   return flag;
}

float findsqrt(float number)
{
   float temp, sq_rt;
   sq_rt = number / 2;
   temp = 0;
   while(sq_rt != temp){
       temp = sq_rt;
       sq_rt = ( number/temp + temp) / 2;
   }
   return sq_rt;
}

output *
exec_1_svc(input *argp, struct svc_req *rqstp)
{
   static output  result;

   int ch = argp->choice;
```

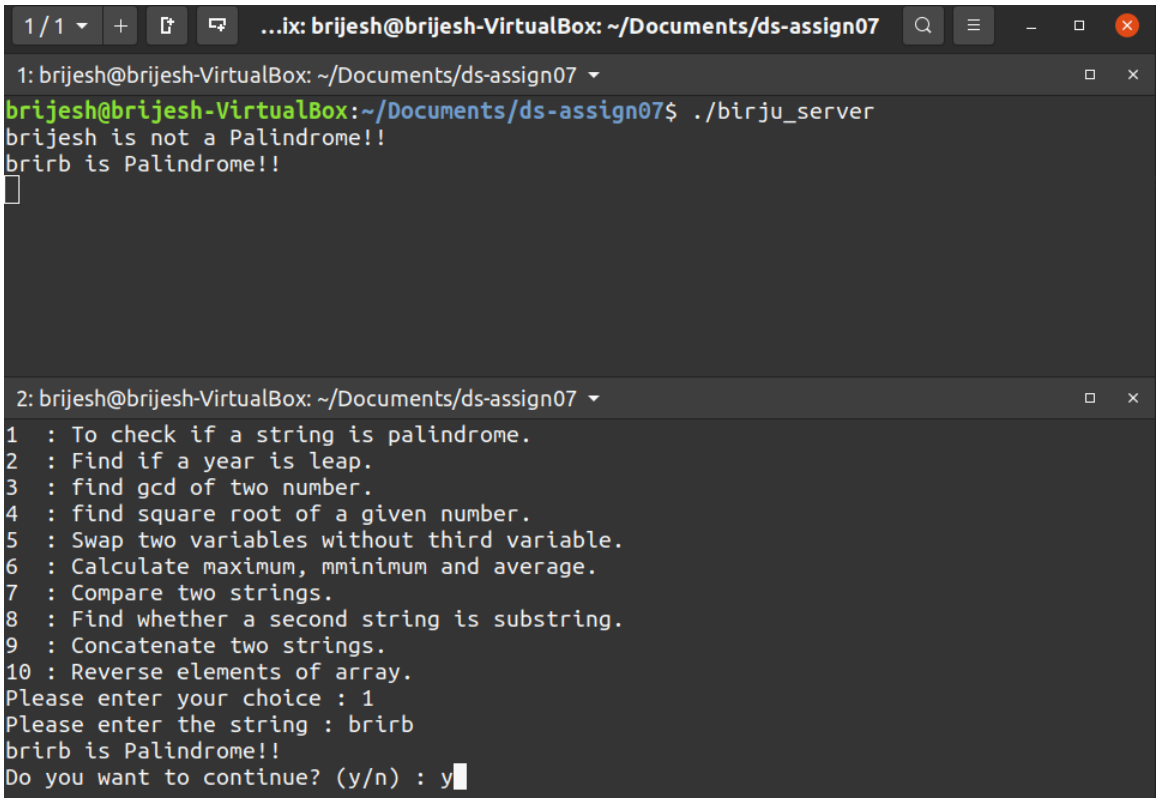
```
switch (ch)
{
case 1:
    result.flag = isPalindrome(argp->str1);
    if(result.flag)
        printf("%s is Palindrome!!\n", argp->str1);
    else
        printf("%s is not a Palindrome!!\n", argp->str1);
    break;
case 2:
    result.flag = isLeap(argp->year);
    if(result.flag)
        printf("%d is a LEAP!!\n", argp->year);
    else
        printf("%d is not a LEAP!!\n", argp->year);
    break;
case 3:
    result.gcd = gcd1(argp->num1, argp->num2);
    printf("GCD[%d,%d] : %d\n", argp->num1, argp->num2, result.gcd);
    break;
case 4:
    result.sqrt = findsqrt(argp->num1);
    printf("Square root of %d : %f\n", argp->num1, result.sqrt);
    break;
case 5:
    result.num1 = argp->num2;
    result.num2 = argp->num1;
    printf("Numebers swapped successfully!!\n");
    break;
case 6:
    result.min = argp->arr[0];
    result.max = argp->arr[0];
    result.avg = argp->arr[0];
    for(int i = 1 ; i < argp->year ; i++)
    {
        if(result.max < argp->arr[i])
        result.max = argp->arr[i];
        if(result.min > argp->arr[i])
        result.min = argp->arr[i];
```

```
            result.avg+=argp->arr[i];
        }
        result.avg/=argp->year;
        printf("MAX : %d", result.max);
        printf("\nMIN : %d", result.min);
        printf("\nAVG : %f\n", result.avg);
        break;
    case 7:
        result.flag = strcmp(argp->str1, argp->str2);
        if(result.flag > 0)
            printf("%s > %s\n", argp->str1, argp->str2);
        else if(result.flag == 0)
            printf("%s = %s\n", argp->str1, argp->str2);
        else
            printf("%s < %s\n", argp->str1, argp->str2);
        break;
    case 8:
        result.flag = isSubstr(argp->str1, argp->str2);
        if(result.flag)
            printf("%s is substring of %s\n", argp->str2, argp->str1);
        else
            printf("%s is not substring of %s\n", argp->str2,
argp->str1);
        break;
    case 9:
        strcpy(result.str,"");
        strcat(result.str,argp->str1);
        strcat(result.str,argp->str2);
        printf("Concatenated string : %s\n", result.str);
        break;
    case 10:
        printf("Original array : ");
        for(int i = 0 ; i < argp->year ; i++)
        {
            printf("%d ", argp->arr[i]);
        }
        printf("\n");
        for(int i = 0 ; i < argp->year ; i++)
        {
```

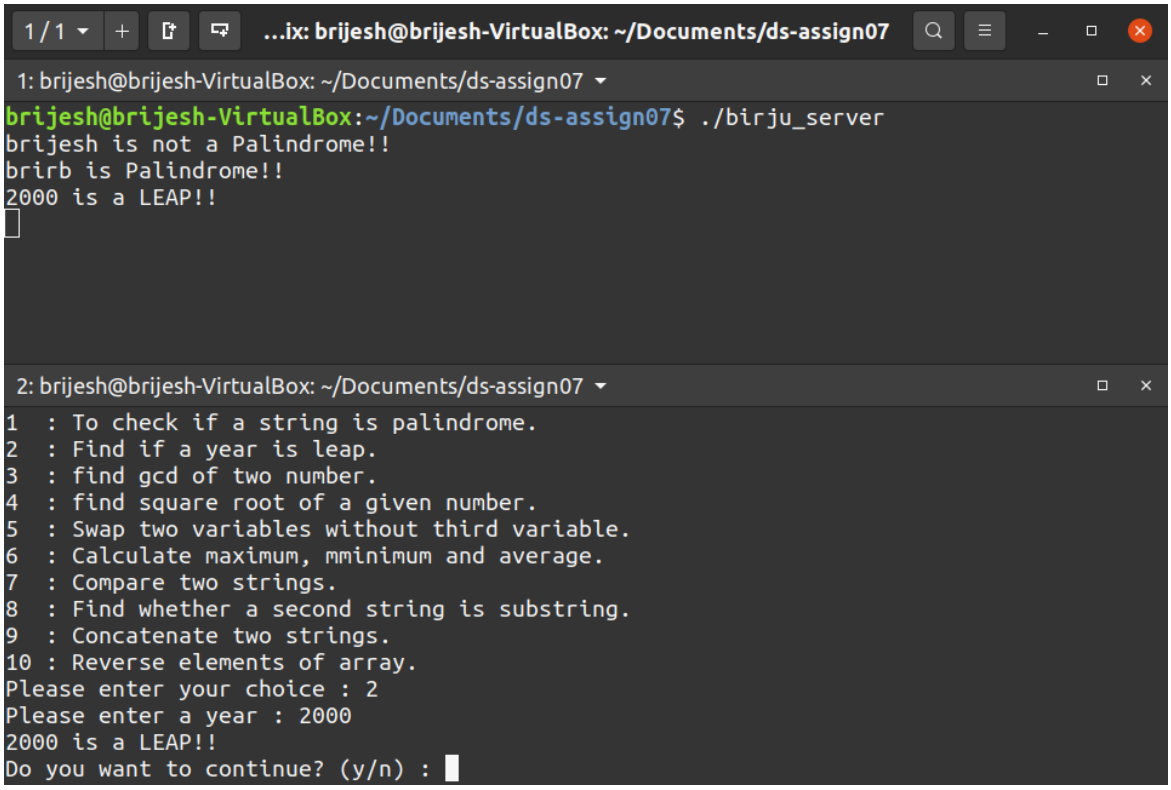
```
            result.arr[i] = argp->arr[argp->year-1-i];
        }
        printf("Reverse array is : ");
        for(int i = 0 ; i < argp->year ; i++)
        {
            printf("%d ",result.arr[i]);
        }
        printf("\n");
        break;
    default:
        printf("Invalid input!!\n");
        break;
    }
    return &result;
}
```

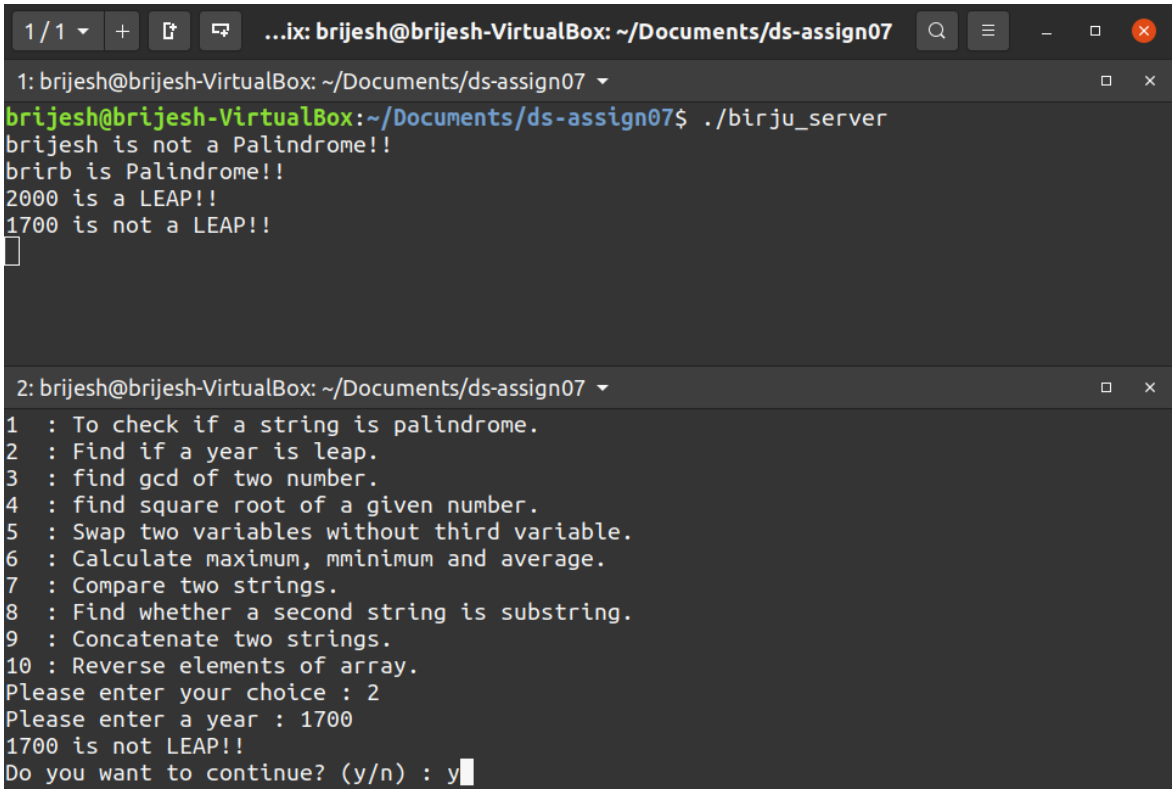
1. String is palindrome or not.



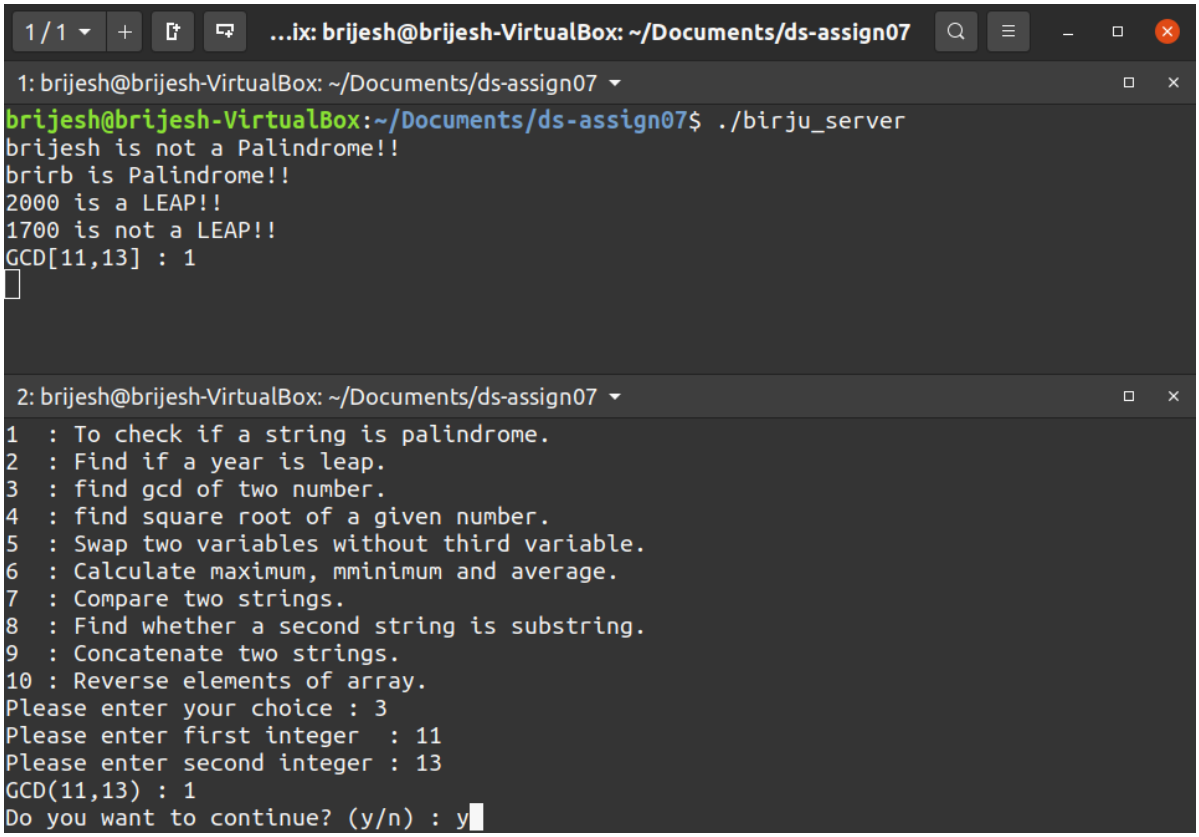


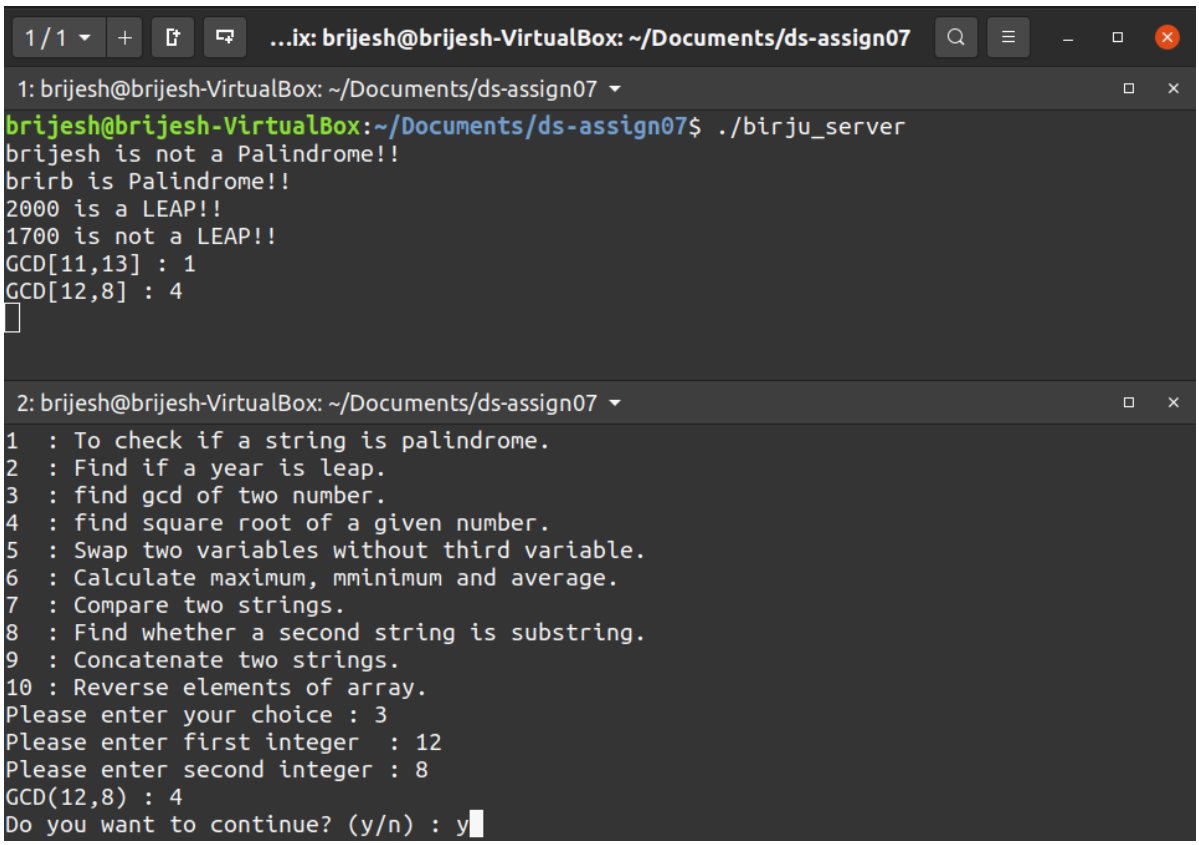
2. Find out if a given year is a Lear Year or not.



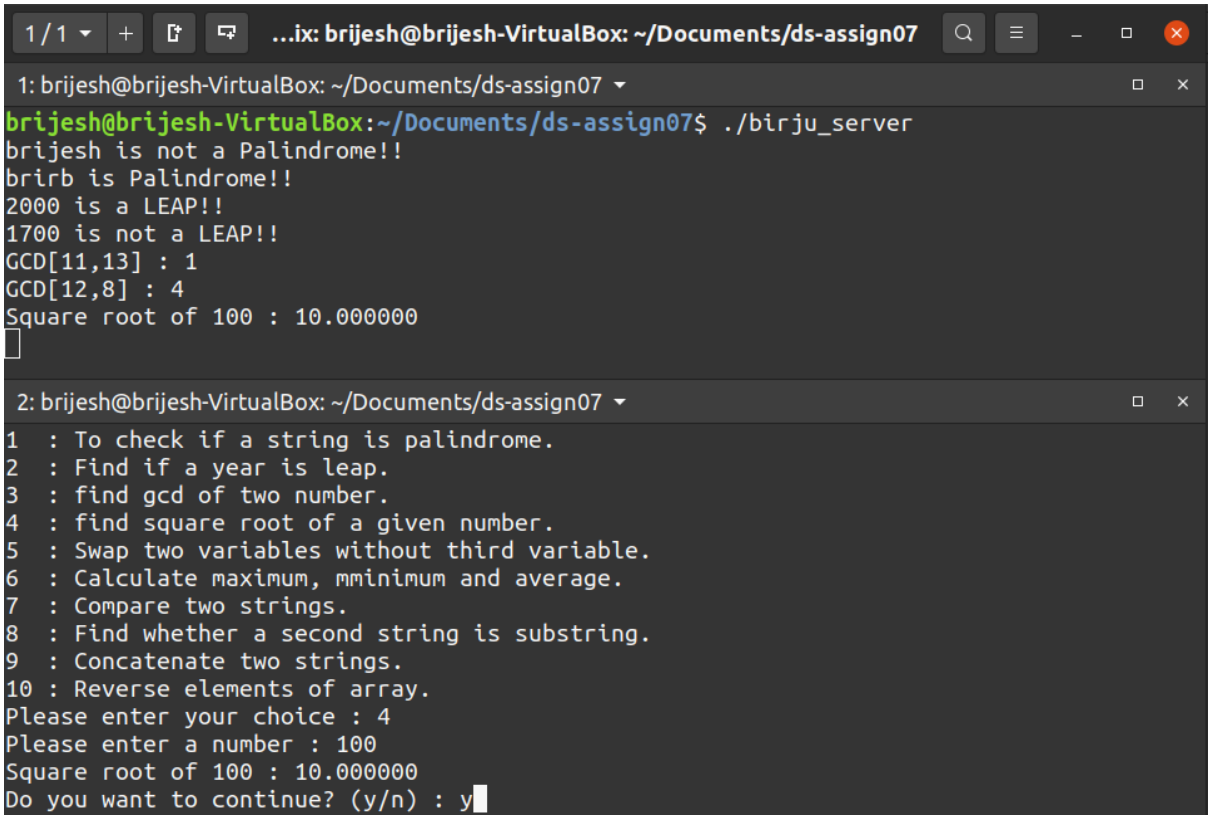


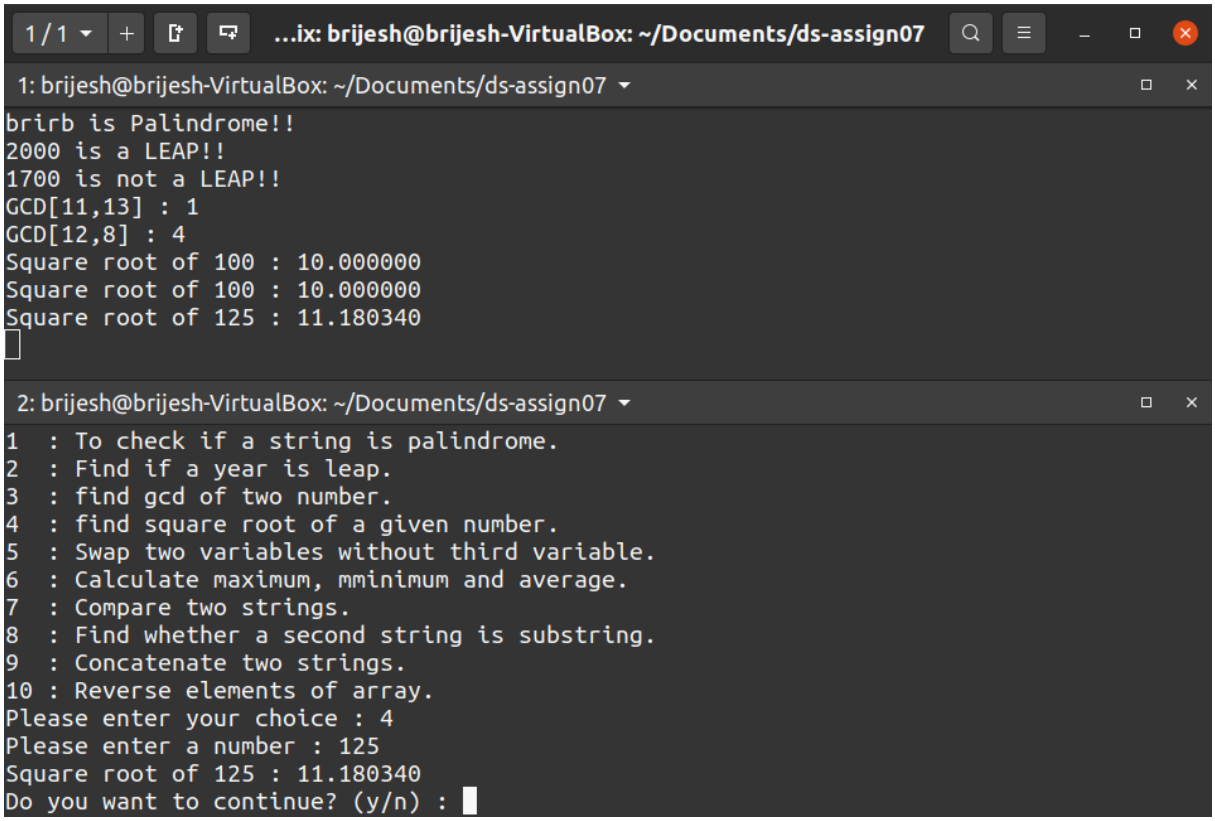
3. Find out the GCD of a given number.



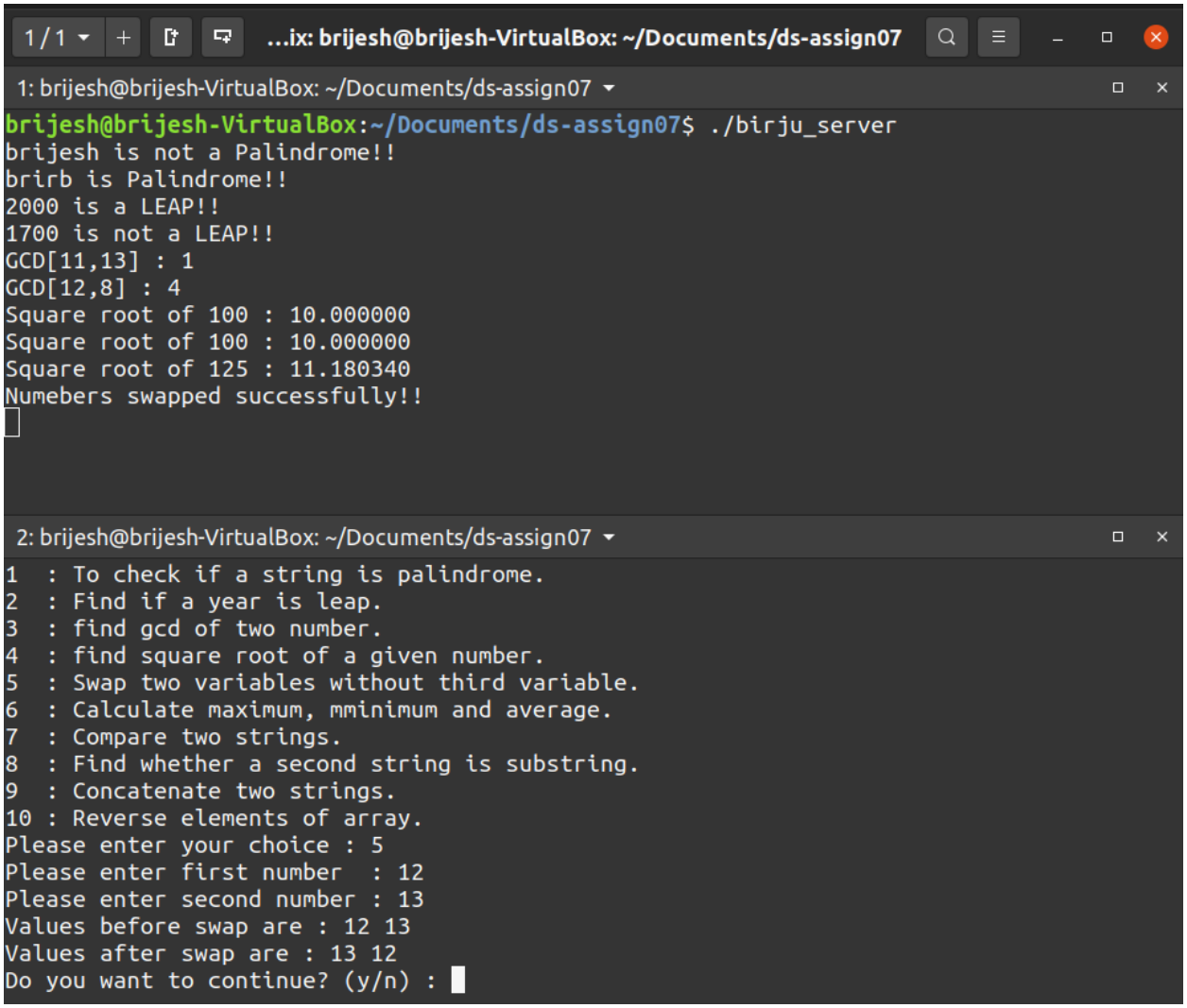


4. Find out the Square root of a given number.

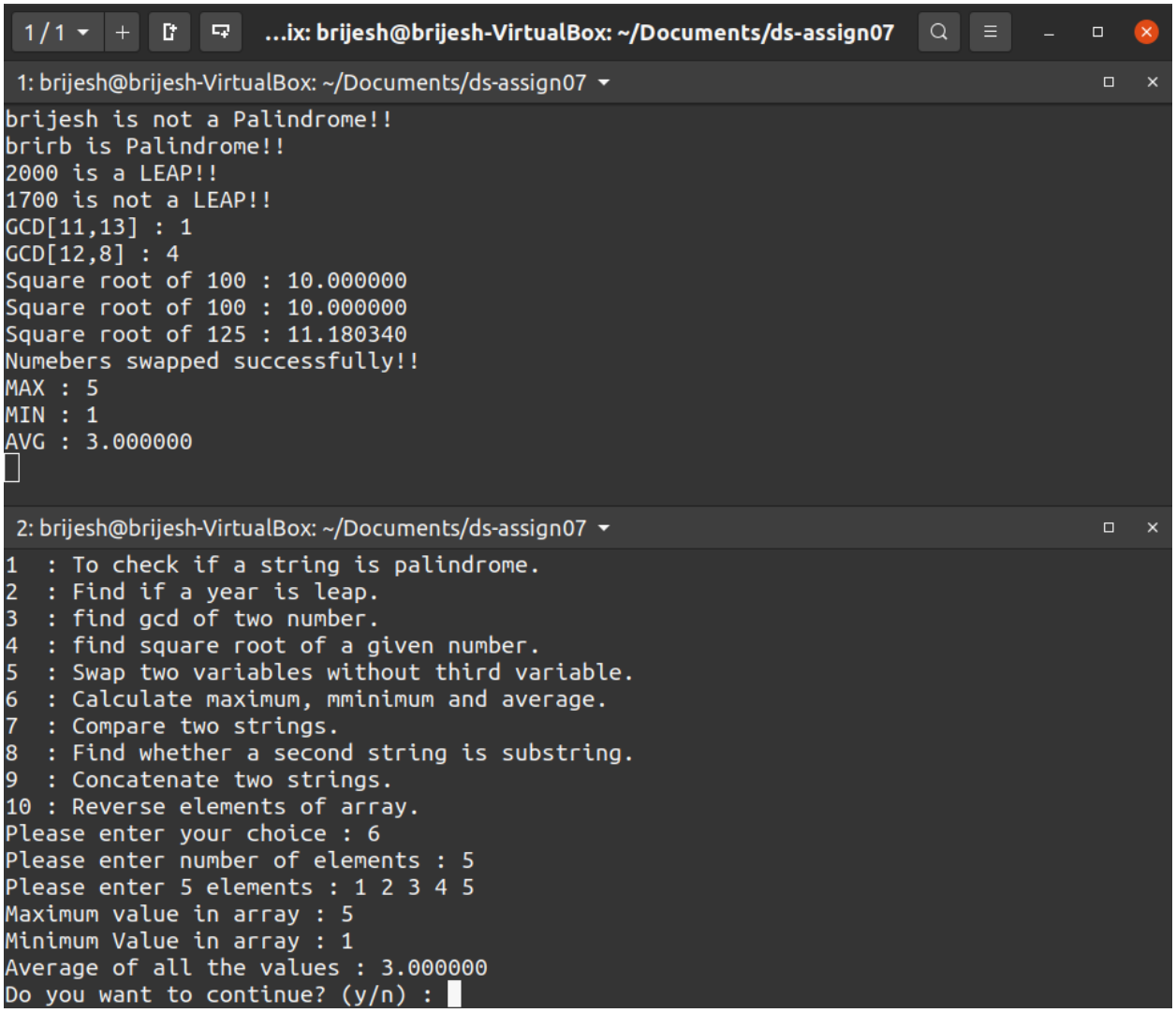




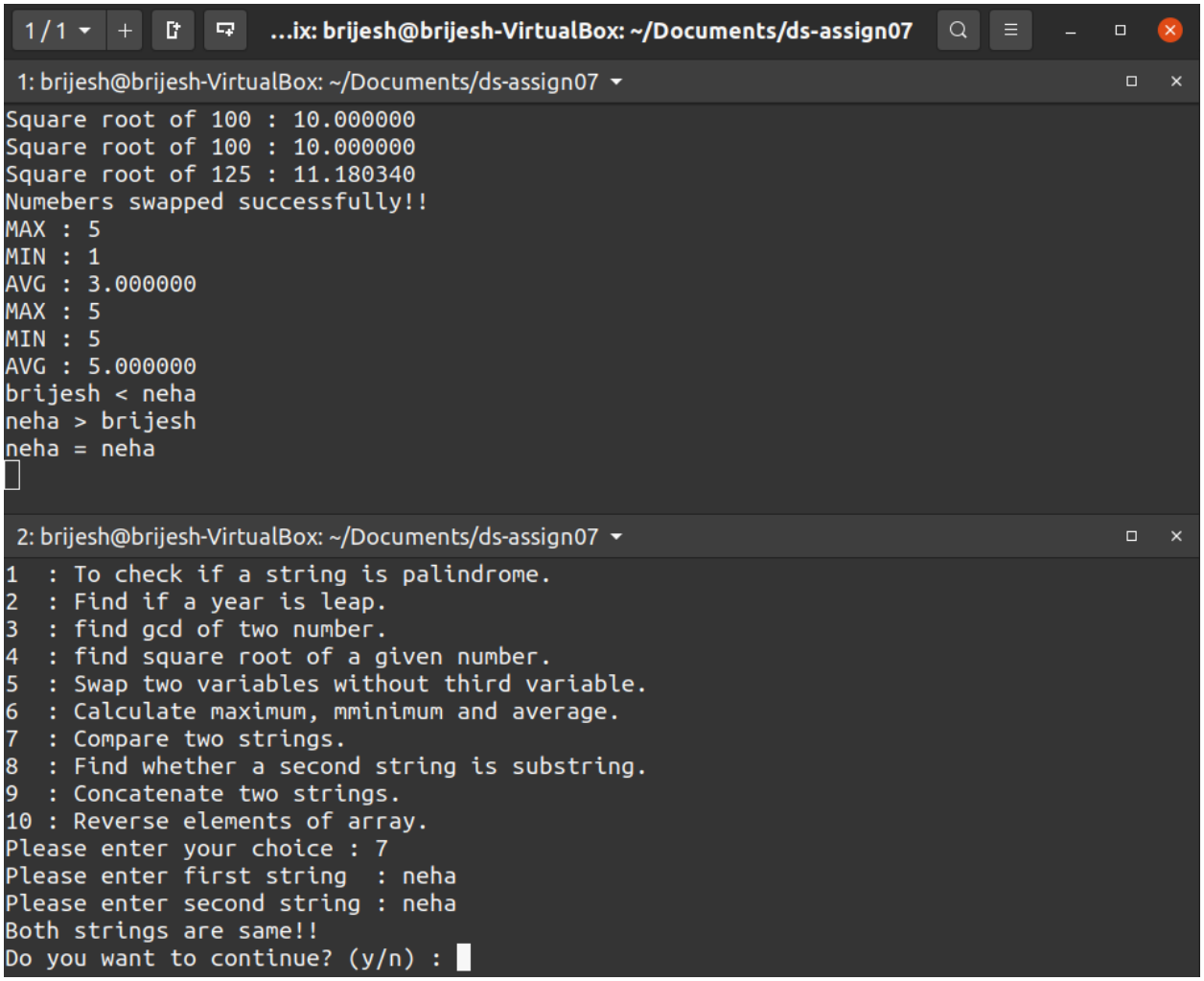
5. Swap two variables without using the 3rd variable.



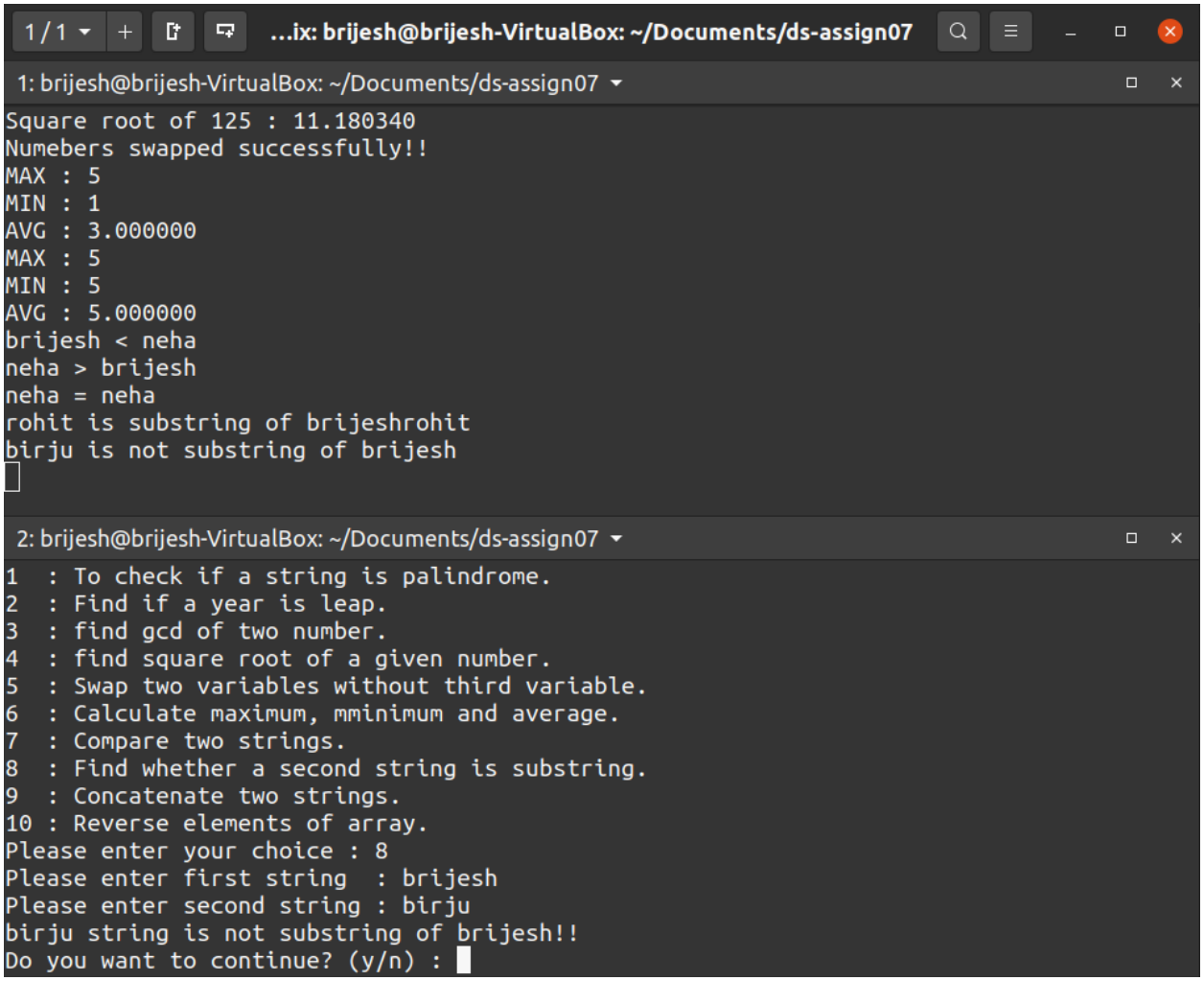
6. Calculate Maximum, Minimum, average of given array.



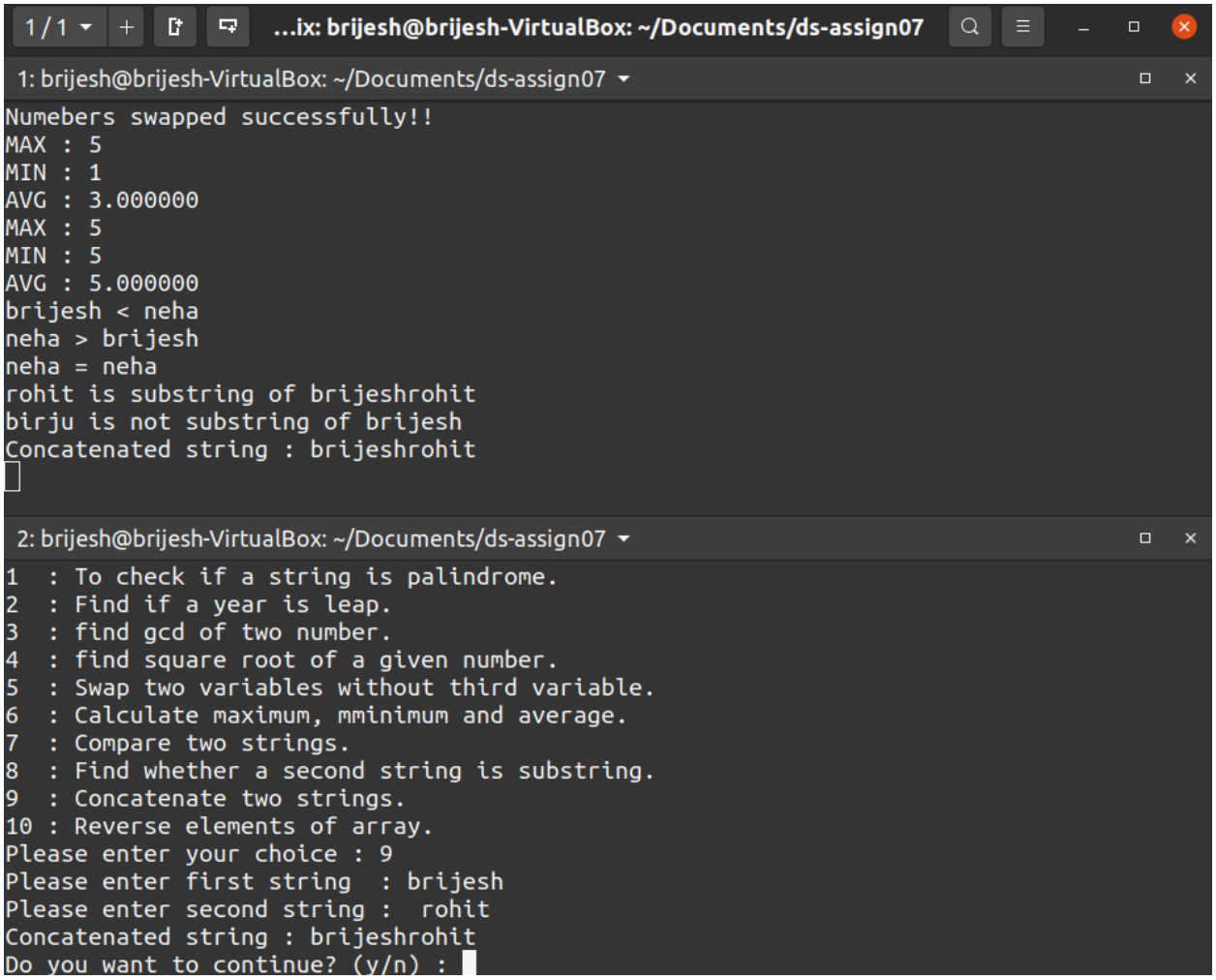
7. Compare the given two strings.



8. Find out whether a given string is substring or not.



9. Concatenate the two strings.



10. Reverse the elements of an array.