Sinhgad Institutes

NBN Sinhgad School of Engineering, Pune

Laboratory Practice-II

Department of Computer Engineering

TE(Computer) **Academic Year: 2021-22**

Prepared By:

Asharani M Chadchankar

(Assistant Professor, Department of Computer Engineering

SINHGAD TECHNICAL EDUCATION SOCIETY'S

NBN SINHGAD SCHOOL OF ENGINEERING

Pune

Department of Computer Engineering

LABORATORY MANUAL

AY: 2021-22

LABORATORY PRACTICE -II

T.E. COMPUTER ENGINEERING
SEMESTER-II

TEACHING SCHEME	CREDIT	EXAMINATION SCHEME
Practical: 4 Hrs / Week	PR: 02	University Term work: 50 Marks
		Practical: 50 Marks

Prepared By:

Asharani M Chadchankar
(Assistant Professor, Department of Computer Engineering)

PREFACE

Laboratory Practice -II

Operating System recommended: 64-bit Open source Linux or its derivative

Programming Languages: C++/JAVA/PYTHON/R

Programming tools recommended: Front End: Java/Perl/PHP/Python/Ruby/.net, **Backend:** Mongo DB/MYSQL/Oracle.

Database Connectivity: ODBC/JDBC, Additional Tools: Octave, Matlab, WEKA**.**

MAIN OBJECTIVES OF LAB:

1. Use tools and techniques in the area of Information Security.

2. Use the knowledge of security for problem solving.

3. Apply the concepts of Information Security to design and develop applications.

CERTIFICATE

This is to certify that Mr. /Miss_____________________________________of class TE Roll No. _______________________Examination Seat No.________________________has completed all the practical work in the Laboratory Practice-II satisfactorily, as prescribed by Savitribai Phule Pune University in the Academic Year 2021-2022.

Staff In-charge **Head of Department** **Principal**

Name of Student	**:**	Omkar Maroti Shinde	**PRN No.**	:72036350M
Student Roll No.	**:**	T5578	**Class**	: TE -2
Subject	**:**	Information Security and Artificial intelligence	**Batch**	:C

I N D E X

Sr.no.	Date	Name of Experiment	Remarks	Signature
		Information Security		
1	14/01/2022	Write a Java/C/C++/Python program that contains a string (char pointer) with a value \Hello World’. The program should AND or and XOR each character in this string with 127 and display the result.		
2	21/01/2022	Write a Java/C/C++/Python program to perform encryption and decryption using the method of Transposition technique.		
3	28/01/2022	Write a Java/C/C++/Python program to implement DES algorithm.		
4	04/02/2022	Write a Java/C/C++/Python program to implement AES Algorithm		
5	28/02/2022	Write a Java/C/C++/Python program to implement RSA algorithm		
		Artificial Intelligence		
6	2/3/2022	Implement depth first search algorithm and Breadth First Search algorithm, Use an undirected graph and develop a recursive algorithm for searching all the vertices of a graph or tree data structure.		

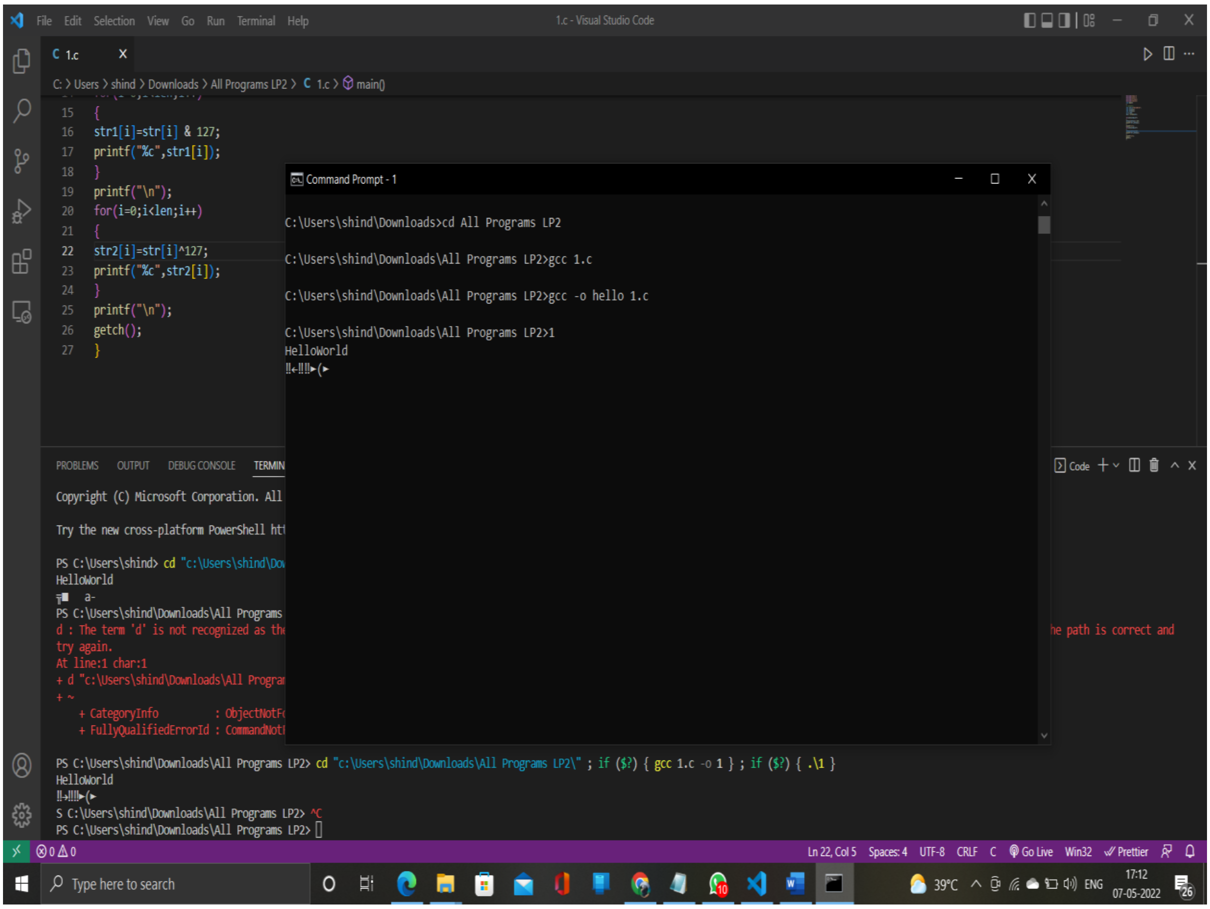
7	9/3/2022	2.Implement A star Algorithm for any game search problem.		
8	30/3/2022	Implement Greedy search algorithm for any of the following application: Prim's Minimal Spanning Tree Algorithm		
9	13/4/2022	Implement a solution for a Constraint Satisfaction Problem using Branch and Bound and Backtracking for n-queens problem or a graph coloring problem		
10	20/4/2022	Develop an elementary chatbot for any suitable customer interaction application.		
11 (mini project)	27/4/2022			

--------------------------- Assignment -1 --

--------------------------------------- XOR a String with a 127 --

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
#include<stdlib.h>
void main()
{
clrscr();
char str[]="HelloWorld";
char str1[11];
char str2[11];
int i,len;
len = strlen(str);

for(i=0;i<len;i++)
{
str1[i]=str[i] & 127;
printf("%c",str1[i]);
}
printf("\n");
for(i=0;i<len;i++)
{
str2[i]=str[i]^127;
printf("%c",str2[i]);
}
printf("\n");
getch();
}
```

Output:



File Edit Selection View Go Run Terminal Help
1.c - Visual Studio Code
1.c
C: > Users > shind > Downloads > All Programs LP2 > 1.c > main()

```
{
str1[i]=str[i] & 127;
printf("%c",str1[i]);
}
printf("\n");
for(i=0;i<len;i++)
{
str2[i]=str[i]^127;
printf("%c",str2[i]);
}
printf("\n");
getch();
}
```

```
C:\Users\shind\Downloads>cd All Programs LP2

C:\Users\shind\Downloads\All Programs LP2>gcc 1.c

C:\Users\shind\Downloads\All Programs LP2>gcc -o hello 1.c

C:\Users\shind\Downloads\All Programs LP2>1
HelloWorld
‼←‼‼►(►
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMIN
Code

```
Copyright (C) Microsoft Corporation. All

Try the new cross-platform PowerShell htt

PS C:\Users\shind> cd "c:\Users\shind\Dow
HelloWorld
╦▀  a-
PS C:\Users\shind\Downloads\All Programs
d : The term 'd' is not recognized as the                                          he path is correct and
try again.
At line:1 char:1
+ d "c:\Users\shind\Downloads\All Progran
+ ~
    + CategoryInfo          : ObjectNotFo
    + FullyQualifiedErrorId : CommandNotf

PS C:\Users\shind\Downloads\All Programs LP2> cd "c:\Users\shind\Downloads\All Programs LP2\" ; if ($?) { gcc 1.c -o 1 } ; if ($?) { .\1 }
HelloWorld
‼→‼‼►(►
S C:\Users\shind\Downloads\All Programs LP2> ^C
PS C:\Users\shind\Downloads\All Programs LP2> 
```

⊗ 0 ⚠ 0
Ln 22, Col 5 Spaces: 4 UTF-8 CRLF C Go Live Win32 Prettier
Type here to search
39°C ENG 17:12 07-05-2022

-- Assignment -2 ---

--- Transposition Technique --

```
 #include<bits/stdc++.h>
 using namespace std; // Key for Columnar Transposition
string const key = "HACK";
 map<int,int> keyMap;
 void setPermutationOrder()
{
// Add the permutation order into map
for(int i=0; i < key.length(); i++)
{
keyMap[key[i]] = i;
}
}
// Encryption
string encryptMessage(string msg)
{
 int row,col,j;
 string cipher = ""; /* calculate column of the matrix*/
 col = key.length();
/* calculate Maximum row of the matrix*/
 row = msg.length()/col;
 if (msg.length() % col)
row += 1;
char matrix[row][col];
for (int i=0,k=0; i < row; i++)
{
```

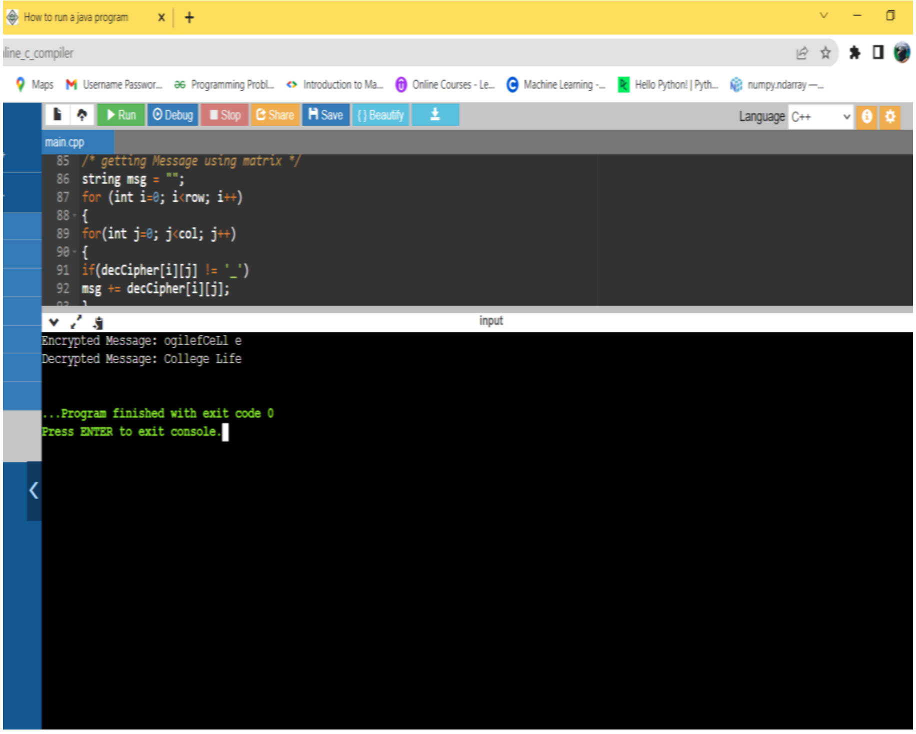
```
for (int j=0; j<col; )

{

if(msg[k] == '\0')

{

/* Adding the padding character '_' */

matrix[i][j] = '_';

j++;

} if( isalpha(msg[k]) || msg[k]==' ')

{

/* Adding only space and alphabet into matrix*/

matrix[i][j] = msg[k];

j++;

}

k++;

}

}

for (map<int,int>::iterator ii = keyMap.begin();

ii!=keyMap.end(); ++ii)

{

j=ii->second;

// getting cipher text from matrix column wise using permuted key

 for (int i=0; i<row; i++)

{ if( isalpha(matrix[i][j]) || matrix[i][j]==' ' || matrix[i][j]=='_')

cipher += matrix[i][j];

}

}

return cipher;

}

// Decryption

string decryptMessage(string cipher)

{
```

```
/* calculate row and column for cipher Matrix */

 int col = key.length();

int row = cipher.length()/col;

char cipherMat[row][col];

/* add character into matrix column wise */

for (int j=0,k=0; j<col; j++)

for (int i=0; i<row; i++)

cipherMat[i][j] = cipher[k++];

/* update the order of key for decryption */

 int index = 0;

for( map<int,int>::iterator ii=keyMap.begin(); ii!=keyMap.end(); ++ii) ii->second = index++;

/* Arrange the matrix column wise according to permutation order by adding
into new matrix */

char decCipher[row][col];

 map<int,int>::iterator ii=keyMap.begin();

int k = 0;

for (int l=0,j; key[l]!='\0'; k++)

{

j = keyMap[key[l++]];

for (int i=0; i<row; i++)

{

decCipher[i][k]=cipherMat[i][j];

}

}

/* getting Message using matrix */

string msg = "";

for (int i=0; i<row; i++)

{

for(int j=0; j<col; j++)

{ if(decCipher[i][j] != '_')

msg +=decCipher[i][j];
}
}
return msg;
}
// Driver Program
 int main(void)
{
/* message */
string msg = "College Life";
setPermutationOrder(); // Calling encryption
function string cipher = encryptMessage(msg);
cout << "Encrypted Message: " << cipher << endl;
// Calling Decryption function

cout << "Decrypted Message: " << decryptMessage(cipher) << endl;

return 0; }
```

Output :

```
Encrypted Message: ogilefCeLl e
```

Decrypted Message: College Life



```
/* getting Message using matrix */
string msg = "";
for (int i=0; i<row; i++)
{
for(int j=0; j<col; j++)
{
if(decCipher[i][j] != '_')
msg += decCipher[i][j];
}
```

```
Encrypted Message: ogilefCeLl e
Decrypted Message: College Life


...Program finished with exit code 0
Press ENTER to exit console.
```

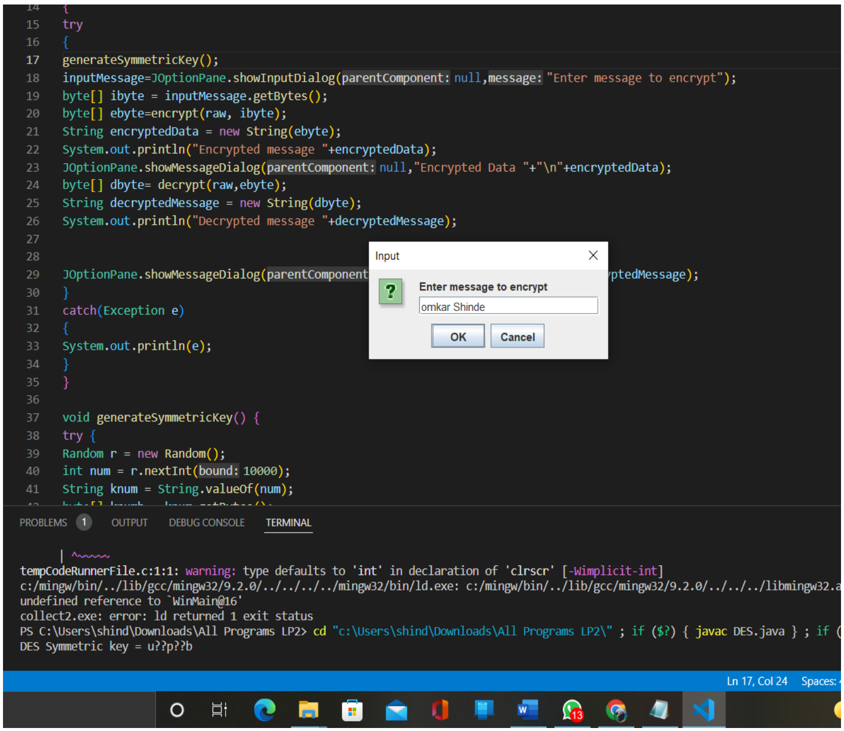
-- Assignment -3 --

-- DES --

```
import javax.swing.*;

import java.security.SecureRandom;

import javax.crypto.Cipher;

import javax.crypto.KeyGenerator;

import javax.crypto.SecretKey;

import javax.crypto.spec.SecretKeySpec;

import java.util.Random ;

class DES {

byte[] skey = new byte[1000];

String skeyString;

static byte[] raw;

String inputMessage,encryptedData,decryptedMessage;

public DES()

{

try

{

generateSymmetricKey();

inputMessage=JOptionPane.showInputDialog(null,"Enter message to encrypt");

byte[] ibyte = inputMessage.getBytes();

byte[] ebyte=encrypt(raw, ibyte);

String encryptedData = new String(ebyte);

System.out.println("Encrypted message "+encryptedData);

JOptionPane.showMessageDialog(null,"Encrypted Data "+"\n"+encryptedData);

byte[] dbyte= decrypt(raw,ebyte);

String decryptedMessage = new String(dbyte);

System.out.println("Decrypted message "+decryptedMessage);



JOptionPane.showMessageDialog(null,"Decrypted Data  "+"\n"+decryptedMessage);
}
catch(Exception e)
{
System.out.println(e);
}
}

void generateSymmetricKey() {
try {
Random r = new Random();
int num = r.nextInt(10000);
String knum = String.valueOf(num);
byte[] knumb = knum.getBytes();
skey=getRawKey(knumb);
skeyString = new String(skey);
System.out.println("DES Symmetric key = "+skeyString);
}
catch(Exception e)
{
System.out.println(e);
}
}
private static byte[] getRawKey(byte[] seed) throws Exception
{
KeyGenerator kgen = KeyGenerator.getInstance("DES");
SecureRandom sr = SecureRandom.getInstance("SHA1PRNG");
sr.setSeed(seed);
kgen.init(56, sr);
SecretKey skey = kgen.generateKey();
raw = skey.getEncoded();
return raw;
```

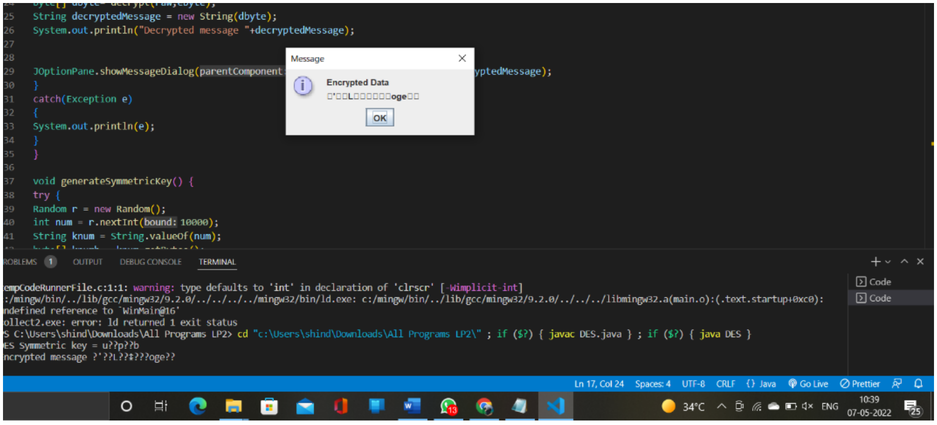
```
}
private static byte[] encrypt(byte[] raw, byte[] clear) throws
Exception {
SecretKeySpec skeySpec = new SecretKeySpec(raw, "DES");
Cipher cipher = Cipher.getInstance("DES");
cipher.init(Cipher.ENCRYPT_MODE, skeySpec);
byte[] encrypted = cipher.doFinal(clear);
return encrypted;
}
private static byte[] decrypt(byte[] raw, byte[] encrypted)
throws Exception
{
SecretKeySpec skeySpec = new SecretKeySpec(raw,
"DES");
Cipher cipher = Cipher.getInstance("DES");
cipher.init(Cipher.DECRYPT_MODE, skeySpec);
byte[] decrypted = cipher.doFinal(encrypted);
return decrypted;
}
public static void main(String args[]) {
DES des = new DES();
}
}
```

Output:



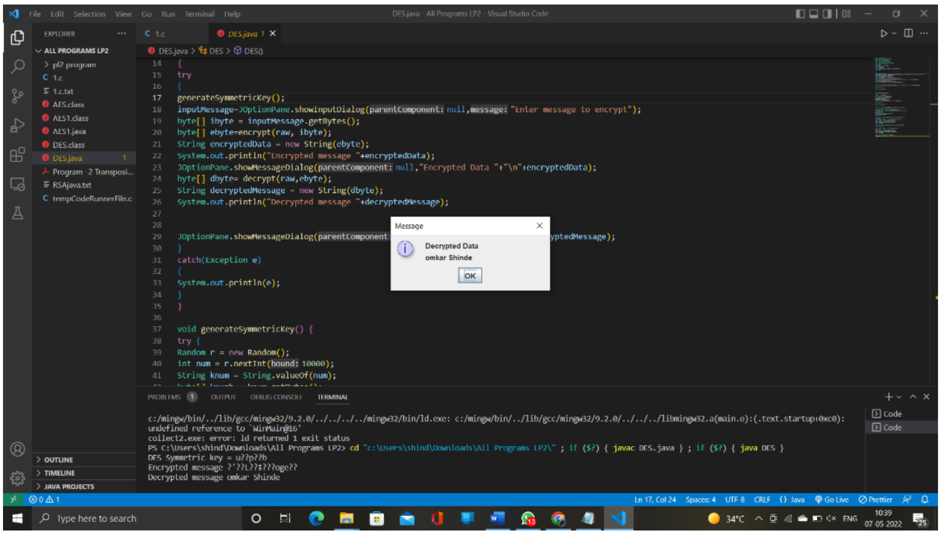
```
{
try
{
generateSymmetricKey();
inputMessage=JOptionPane.showInputDialog(parentComponent: null,message: "Enter message to encrypt");
byte[] ibyte = inputMessage.getBytes();
byte[] ebyte=encrypt(raw, ibyte);
String encryptedData = new String(ebyte);
System.out.println("Encrypted message "+encryptedData);
JOptionPane.showMessageDialog(parentComponent: null,"Encrypted Data "+"\n"+encryptedData);
byte[] dbyte= decrypt(raw,ebyte);
String decryptedMessage = new String(dbyte);
System.out.println("Decrypted message "+decryptedMessage);


JOptionPane.showMessageDialog(parentComponent                    yptedMessage);
}
catch(Exception e)
{
System.out.println(e);
}
}

void generateSymmetricKey() {
try {
Random r = new Random();
int num = r.nextInt(bound: 10000);
String knum = String.valueOf(num);
```

```
tempCodeRunnerFile.c:1:1: warning: type defaults to 'int' in declaration of 'clrscr' [-Wimplicit-int]
c:/mingw/bin/../lib/gcc/mingw32/9.2.0/../../../../mingw32/bin/ld.exe: c:/mingw/bin/../lib/gcc/mingw32/9.2.0/../../../libmingw32.a
undefined reference to `WinMain@16'
collect2.exe: error: ld returned 1 exit status
PS C:\Users\shind\Downloads\All Programs LP2> cd "c:\Users\shind\Downloads\All Programs LP2\" ; if ($?) { javac DES.java } ; if (
DES Symmetric key = u??p??b
```



```
String decryptedMessage = new String(dbyte);
System.out.println("Decrypted message "+decryptedMessage);


JOptionPane.showMessageDialog(parentComponent                    yptedMessage);
}
catch(Exception e)
{
System.out.println(e);
}
}

void generateSymmetricKey() {
try {
Random r = new Random();
int num = r.nextInt(bound: 10000);
String knum = String.valueOf(num);
```

```
empCodeRunnerFile.c:1:1: warning: type defaults to 'int' in declaration of 'clrscr' [-Wimplicit-int]
:/mingw/bin/../lib/gcc/mingw32/9.2.0/../../../../mingw32/bin/ld.exe: c:/mingw/bin/../lib/gcc/mingw32/9.2.0/../../../libmingw32.a(main.o):(.text.startup+0xc0):
ndefined reference to `WinMain@16'
ollect2.exe: error: ld returned 1 exit status
S C:\Users\shind\Downloads\All Programs LP2> cd "c:\Users\shind\Downloads\All Programs LP2\" ; if ($?) { javac DES.java } ; if ($?) { java DES }
ES Symmetric key = u??p??b
ncrypted message ?'??L??‡???oge??
```



```
{
try
{
generateSymmetricKey();
inputMessage=JOptionPane.showInputDialog(parentComponent: null,message: "Enter message to encrypt");
byte[] ibyte = inputMessage.getBytes();
byte[] ebyte=encrypt(raw, ibyte);
String encryptedData = new String(ebyte);
System.out.println("Encrypted message "+encryptedData);
JOptionPane.showMessageDialog(parentComponent: null,"Encrypted Data "+"\n"+encryptedData);
byte[] dbyte= decrypt(raw,ebyte);
String decryptedMessage = new String(dbyte);
System.out.println("Decrypted message "+decryptedMessage);


JOptionPane.showMessageDialog(parentComponent                    yptedMessage);
}
catch(Exception e)
{
System.out.println(e);
}
}

void generateSymmetricKey() {
try {
Random r = new Random();
int num = r.nextInt(bound: 10000);
String knum = String.valueOf(num);
```

```
c:/mingw/bin/../lib/gcc/mingw32/9.2.0/../../../../mingw32/bin/ld.exe: c:/mingw/bin/../lib/gcc/mingw32/9.2.0/../../../libmingw32.a(main.o):(.text.startup+0xc0):
undefined reference to `WinMain@16'
collect2.exe: error: ld returned 1 exit status
PS C:\Users\shind\Downloads\All Programs LP2> cd "c:\Users\shind\Downloads\All Programs LP2\" ; if ($?) { javac DES.java } ; if ($?) { java DES }
DES Symmetric key = u??p??b
Encrypted message ?'??L??‡???oge??
Decrypted message omkar Shinde
```

--- Assignment-4 ---

--- 4 Aes1 ---

```
import base64
import hashlib
from Crypto import Random
from Crypto.Cipher import AES

class AESCipher(object):

    def __init__(self, key):
        self.bs = AES.block_size
        self.key = hashlib.sha256(key.encode()).digest()

    def encrypt(self, raw):
        raw = self._pad(raw)
       i iv = Random.new().read(AES.block_size)
        cipher = AES.new(self.key, AES.MODE_CBC, iv)
        return base64.b64encode(iv + cipher.encrypt(raw.encode()))

    def decrypt(self, enc):
        enc = base64.b64decode(enc)
        iv = enc[:AES.block_size]
        cipher = AES.new(self.key, AES.MODE_CBC, iv)
        return self._unpad(cipher.decrypt(enc[AES.block_size:])).decode('utf-8')

    def _pad(self, s):
        return s + (self.bs - len(s) % self.bs) * chr(self.bs - len(s) % self.bs)

    @staticmethod
    def _unpad(s):
```

```
    return s[:-ord(s[len(s)-1:])]

obj = AESCipher("Sixteen byte key")
e = obj.encrypt("Sixteen byte key")
print("Encrypted cipher text : ", e)
d = obj.decrypt(e)
print("Decrypted cipher text : ", d)
```

Output:

```
PlainText=College Life
EncryptedText=Gxjm8EDMrwHcT15PuAd4g==
DecryptedText=Collge Life
```

-- Assignment-5 --

-- RSA ---

```
import java.math.*;

import java.util.Random;

import java.util.Scanner;


public class RSA {

   static Scanner sc = new Scanner(System.in);

   public static void main(String[ ]args) {

      System.out.print("Enter a Prime number: ");

      BigInteger p=sc.nextBigInteger(); //Here's one prime number..

      System.out.print("Enter another prime number: ");

      BigInteger q=sc.nextBigInteger(); //..and another.

      BigInteger n=p.multiply(q);

      BigInteger n2=p.subtract(BigInteger.ONE).multiply(q.subtract(BigInteger.ONE));

      BigInteger e = generateE(n2);

      BigInteger d = e.modInverse(n2); // Here's the multiplicative inverse

      System.out.println("Encryption keys are: " + e + ", " + n);

      System.out.println("Decryption keys are: " + d + ", " + n);

   }




   private static BigInteger generateE(BigInteger fiofn){

      int y, intGCD;
```

```
    Random x = new Random();
    BigInteger e;

    do{
      y= x.nextInt(fiofn.intValue()-1);
      String z = Integer.toString(y);
      e = new BigInteger(z);
      BigInteger gcd = fiofn.gcd(e);
      intGCD = gcd.intValue();
    }

    while(y<=2 || intGCD!=1);
    return e;
  }
}
```

Out put:

```
Enter a Prime number: 5
Enter another prime number: 7
Encryption keys are: 13, 35
Decryption keys are: 13, 35
```

---AI Assignment-1---

---BFS--

```
graph = {
    'A': ['B', 'C', "D"],
    'B': ['E', "F"],
    'C': ['G', "I"],
    'D': ["I"],
    'E': [],
    "F": [],
    'G': [],
    "I": []
}

def bfs(visit_complete, graph, current_node):
    visit_complete.append(current_node)
    queue = []
    queue.append(current_node)

    while queue:
        s = queue.pop(0)
        print(s)

        for neighbour in graph[s]:
            if neighbour not in visit_complete:
                visit_complete.append(neighbour)
                queue.append(neighbour)

bfs([], graph, 'A')
```

output:

```
A

B

C

D

E

F

G

I
```

---DFS---

```python
# Using a Python dictionary to act as an adjacency list
graph = {
  '5' : ['3','7'],
  '3' : ['2', '4'],
  '7' : ['8'],
  '2' : [],
  '4' : ['8'],
  '8' : []
}

visited = set() # Set to keep track of visited nodes of graph.

def dfs(visited, graph, node):  #function for dfs
    if node not in visited:
        print (node)
        visited.add(node)
        for neighbour in graph[node]:
            dfs(visited, graph, neighbour)

# Driver Code
print("Following is the Depth-First Search")
dfs(visited, graph, '5')
```

output:

```
Following is the Depth-First Search

5

3

2

4

8

7
```

--Assignment -2---

---A * Search---

```
class Node:
 def __init__(self, data, level, fval):


 self.data = data
 self.level = level
 self.fval = fval


 def generate_child(self):


 x, y = self.find(self.data, '_')
 """ val_list contains position values for moving the blank space in either
of  the 4 directions [up,down,left,right] respectively. """  val_list = [[x, y -
1], [x, y + 1], [x - 1, y], [x + 1, y]]
 children = []
 for i in val_list:
 child = self.shuffle(self.data, x, y, i[0], i[1])
 if child is not None:
 child_node = Node(child, self.level + 1, 0)
 children.append(child_node)
 return children


 def shuffle(self, puz, x1, y1, x2, y2):
```

```
 """ Move the blank space in the given direction and if the position value are
out  of limits the return None """
 if x2 >= 0 and x2 < len(self.data) and y2 >= 0 and y2 <
len(self.data):  temp_puz = []
 temp_puz = self.copy(puz)
 temp = temp_puz[x2][y2]
 temp_puz[x2][y2] = temp_puz[x1][y1]
 temp_puz[x1][y1] = temp
 return temp_puz
 else:
 return None


 def copy(self, root):
 """ Copy function to create a similar matrix of the given
node"""  temp = []
 for i in root:
 t = []
 for j in i:
 t.append(j)
 temp.append(t)
 return temp


 def find(self, puz, x):
```

```
 """ Specifically used to find the position of the blank space

"""  for i in range(0, len(self.data)):

 for j in range(0, len(self.data)):

 if puz[i][j] == x:

 return i, j




class Puzzle:

 def __init__(self, size):

 """ Initialize the puzzle size by the specified size,open and closed lists to empty

"""  self.n = size

 self.open = []

 self.closed = []


 def accept(self):

 """ Accepts the puzzle from the user """

 puz = []

 for i in range(0, self.n):
 temp = input().split(" ")

 puz.append(temp)

 return puz


 def f(self, start, goal):
```

```
""" Heuristic Function to calculate hueristic value f(x) = h(x) + g(x)
"""  return self.h(start.data, goal) + start.level

def h(self, start, goal):
""" Calculates the different between the given puzzles
"""  temp = 0
for i in range(0, self.n):
for j in range(0, self.n):
if start[i][j] != goal[i][j] and start[i][j] != '_':
temp += 1
return temp

def process(self):
""" Accept Start and Goal Puzzle state"""
print("Enter the start state matrix \n")
start = self.accept()
print("Enter the goal state matrix \n")
goal = self.accept()
start = Node(start, 0, 0)
start.fval = self.f(start, goal)
""" Put the start node in the open list"""
self.open.append(start)
print("\n\n")
while True:
cur = self.open[0]

print("")

print(" | ")

print(" | ")

print(" \\\'/ \n")

for i in cur.data:

for j in i:

print(j, end=" ")

print("")

""" If the difference between current and goal node is 0 we have reached the
goal  node"""

if (self.h(cur.data, goal) == 0):

break

for i in cur.generate_child():

i.fval = self.f(i, goal)

self.open.append(i)
self.closed.append(cur)

del self.open[0]


""" sort the opne list based on f value

"""  self.open.sort(key=lambda x: x.fval,

reverse=False)


puz = Puzzle(3)
```

```
puz.process()
```

-

OUTPUT:-

```
Enter the start state matrix


1 2 3

_ 4 6

7 5 8

Enter the goal state matrix


1 2 3

4 5 6

7 8 _
 |

 |

 \'/


1

2

3



_
```

4

6

7

5

8

|

|

\'/

1

2

3

4

_

6

7

5

8

|

|

\'/

1

2

3

4

5

6

7

_

8

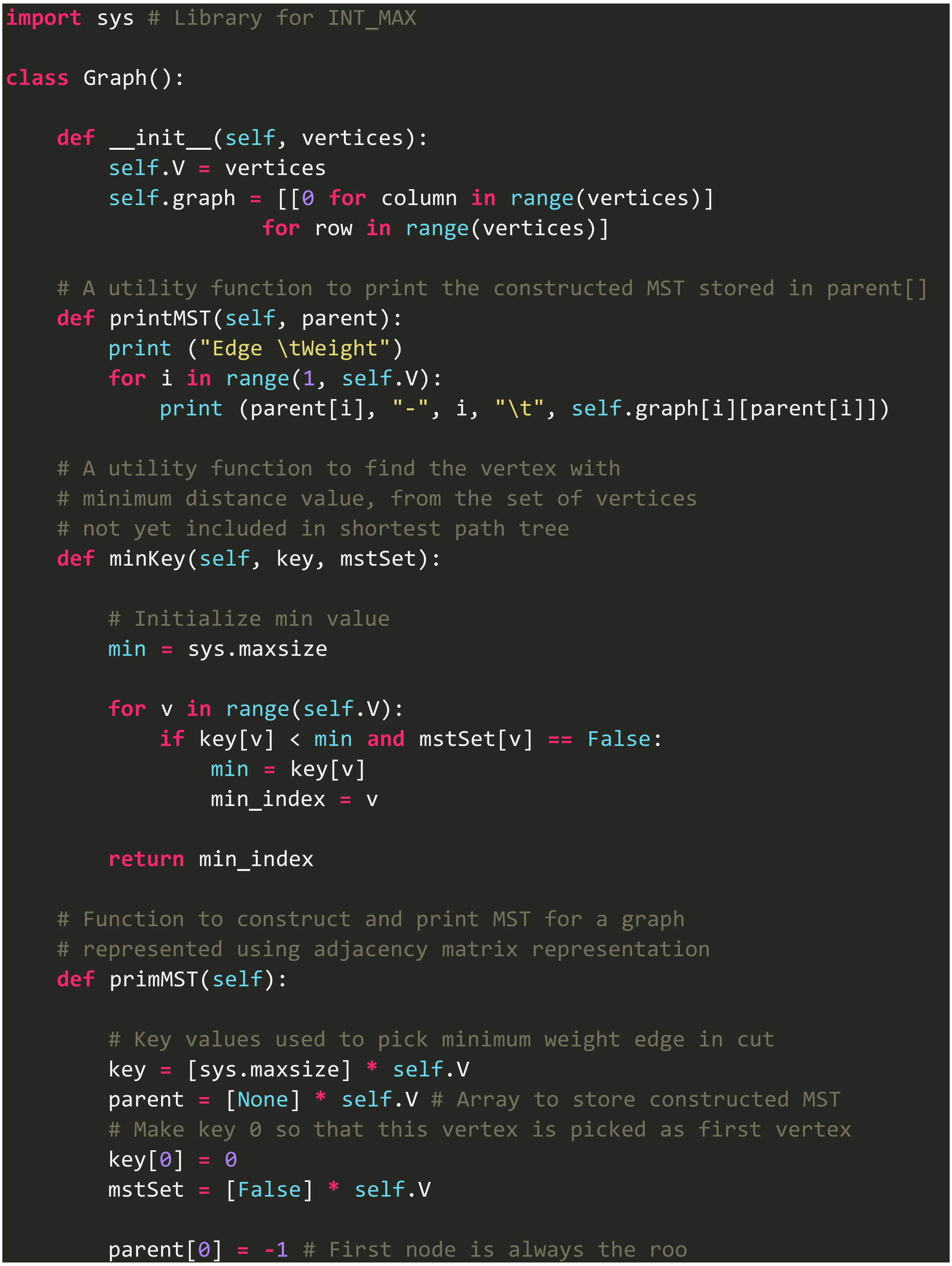
|

|

\'/

--Assignment -3--

-----------------------------------prime’s Algorithm Minimum Spanning Spaning tree-----------------------------



```
import sys # Library for INT_MAX

class Graph():

    def __init__(self, vertices):
        self.V = vertices
        self.graph = [[0 for column in range(vertices)]
                    for row in range(vertices)]

    # A utility function to print the constructed MST stored in parent[]
    def printMST(self, parent):
        print ("Edge \tWeight")
        for i in range(1, self.V):
            print (parent[i], "-", i, "\t", self.graph[i][parent[i]])

    # A utility function to find the vertex with
    # minimum distance value, from the set of vertices
    # not yet included in shortest path tree
    def minKey(self, key, mstSet):

        # Initialize min value
        min = sys.maxsize

        for v in range(self.V):
            if key[v] < min and mstSet[v] == False:
                min = key[v]
                min_index = v

        return min_index

    # Function to construct and print MST for a graph
    # represented using adjacency matrix representation
    def primMST(self):

        # Key values used to pick minimum weight edge in cut
        key = [sys.maxsize] * self.V
        parent = [None] * self.V # Array to store constructed MST
        # Make key 0 so that this vertex is picked as first vertex
        key[0] = 0
        mstSet = [False] * self.V

        parent[0] = -1 # First node is always the roo
        for cout in range(self.V):

            # Pick the minimum distance vertex from
            # the set of vertices not yet processed.
            # u is always equal to src in first iteration
            u = self.minKey(key, mstSet)

            # Put the minimum distance vertex in
            # the shortest path tree
            mstSet[u] = True

            # Update dist value of the adjacent vertices
            # of the picked vertex only if the current
            # distance is greater than new distance and
            # the vertex in not in the shortest path tree
            for v in range(self.V):

                # graph[u][v] is non zero only for adjacent vertices of m
                # mstSet[v] is false for vertices not yet included in MST
                # Update the key only if graph[u][v] is smaller than
key[v]
                if self.graph[u][v] > 0 and mstSet[v] == False and
key[v] > self.graph[u][v]:
                        key[v] = self.graph[u][v]
                        parent[v] = u

        self.printMST(parent)

g = Graph(5)
g.graph = [ [0, 2, 0, 6, 0],
            [2, 0, 3, 8, 5],
            [0, 3, 0, 0, 7],
            [6, 8, 0, 0, 9],
            [0, 5, 7, 9, 0]]

g.primMST();
```

output:

```
Edge   Weight
0 - 1    2
1 - 2    3
0 - 3    6
1 - 4    5
```

--Assignment 4--

---n Queen ---

N queen problem

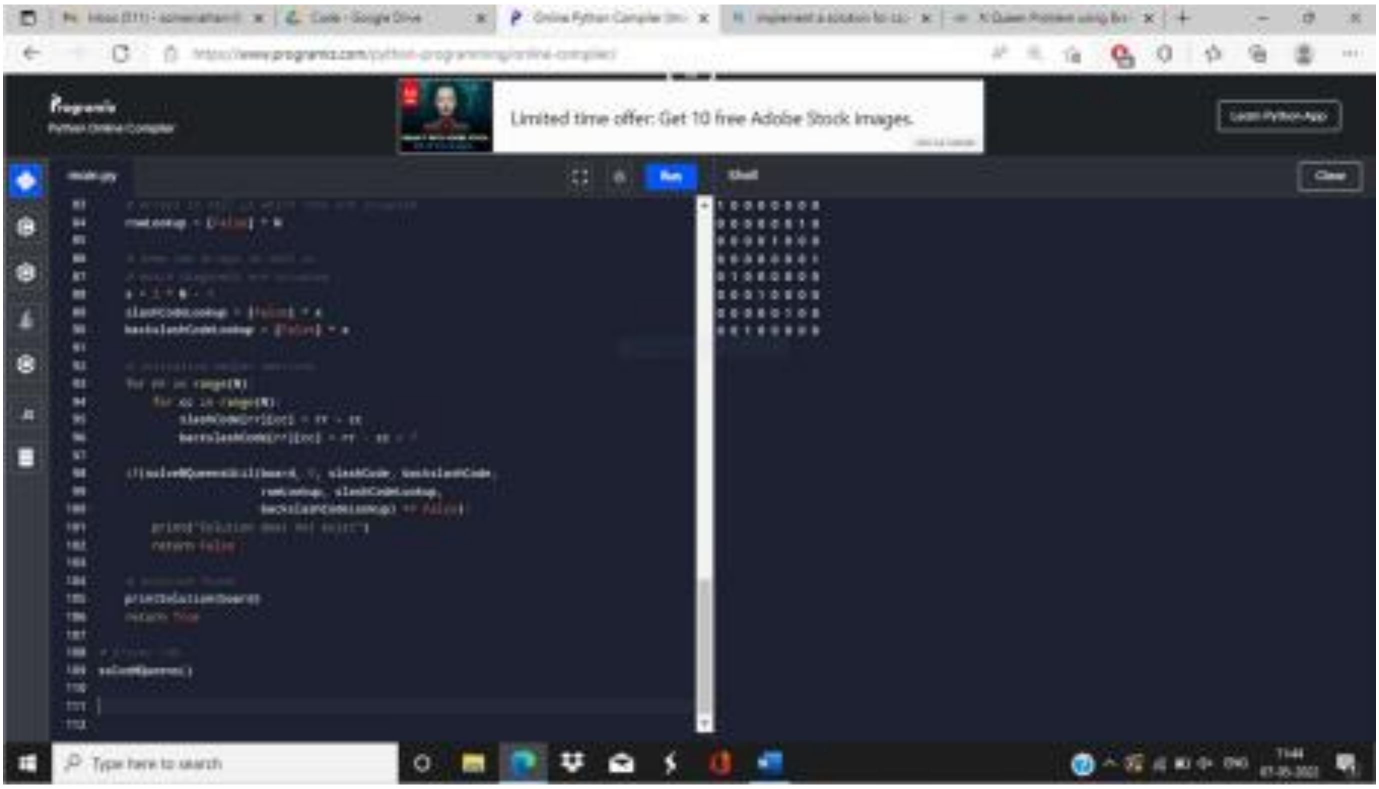
```
""" Python3 program to solve N Queen Problem
using Branch or Bound """

N = 8

""" A utility function to print solution """
def printSolution(board):
 for i in range(N):
 for j in range(N):
 print(board[i][j], end = " ")   print()

""" A Optimized function to check if
a queen can be placed on board[row][col] """
def isSafe(row, col, slashCode, backslashCode,
 rowLookup, slashCodeLookup,
 backslashCodeLookup):   if
(slashCodeLookup[slashCode[row][col]]
or   backslashCodeLookup[backslashCode[row][col]]
or   rowLookup[row]):
 return False
 return True
""" A recursive utility function
to solve N Queen problem """
```

```
def solveNQueensUtil(board, col, slashCode, backslashCode,
 rowLookup, slashCodeLookup,
 backslashCodeLookup):

 """ base case: If all queens are
 placed then return True """
 if(col >= N):
 return True
 for i in range(N):
 if(isSafe(i, col, slashCode, backslashCode,
 rowLookup, slashCodeLookup,
 backslashCodeLookup)):

 """ Place this queen in board[i][col] """
 board[i][col] = 1
 rowLookup[i] = True
 slashCodeLookup[slashCode[i][col]] = True
 backslashCodeLookup[backslashCode[i][col]] = True
 """ recur to place rest of the queens """
 if(solveNQueensUtil(board, col + 1,
 slashCode, backslashCode,   rowLookup,
slashCodeLookup,   backslashCodeLookup)):   return True


 """ If placing queen in board[i][col]
 doesn't lead to a solution,then backtrack """

 """ Remove queen from board[i][col] """   board[i][col] = 0
 rowLookup[i] = False
 slashCodeLookup[slashCode[i][col]] =
False   backslashCodeLookup[backslashCode[i][col]] = False
 """ If queen can not be place in any row in
 this column col then return False """
 return False

""" This function solves the N Queen problem using
Branch or Bound. It mainly uses solveNQueensUtil()to
solve the problem. It returns False if queens
cannot be placed,otherwise return True or
prints placement of queens in the form of 1s.
Please note that there may be more than one
solutions,this function prints one of the
feasible solutions."""
def solveNQueens():
 board = [[0 for i in range(N)]
 for j in range(N)]

 # helper matrices
 slashCode = [[0 for i in range(N)]
 for j in range(N)]
 backslashCode = [[0 for i in range(N)]
 for j in range(N)]
 # arrays to tell us which rows are occupied
 rowLookup = [False] * N

 # keep two arrays to tell us
 # which diagonals are occupied
 x = 2 * N - 1
 slashCodeLookup = [False] * x
 backslashCodeLookup = [False] * x

 # initialize helper matrices
 for rr in range(N):
```

```
 for cc in range(N):
 slashCode[rr][cc] = rr + cc
 backslashCode[rr][cc] = rr - cc + 7

 if(solveNQueensUtil(board, 0, slashCode, backslashCode,   rowLookup,
slashCodeLookup,   backslashCodeLookup) == False):   print("Solution
does not exist")
 return False

 # solution found
 printSolution(board)
 return True

# Driver Cde
solveNQueens()
```

output

```
1 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0
0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0
0 0 0 1 0 0 0 0
0 0 0 0 0 1 0 0
0 0 1 0 0 0 0 0
```



```
    # arrays to tell us which rows are occupied
    rowLookup = [False] * N

    # keep two arrays to tell us
    # which diagonals are occupied
    x = 2 * N - 1
    slashCodeLookup = [False] * x
    backslashCodeLookup = [False] * x

    # initialize helper matrices
    for rr in range(N):
        for cc in range(N):
            slashCode[rr][cc] = rr + cc
            backslashCode[rr][cc] = rr - cc + 7

    if(solveNQueensUtil(board, 0, slashCode, backslashCode,
                        rowLookup, slashCodeLookup,
                        backslashCodeLookup) == False):
        print("Solution does not exist")
        return False

    # solution found
    printSolution(board)
    return True

# Driver Code
solveNQueens()
```

```
1 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0
0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0
0 0 0 1 0 0 0 0
0 0 0 0 0 1 0 0
0 0 1 0 0 0 0 0
```

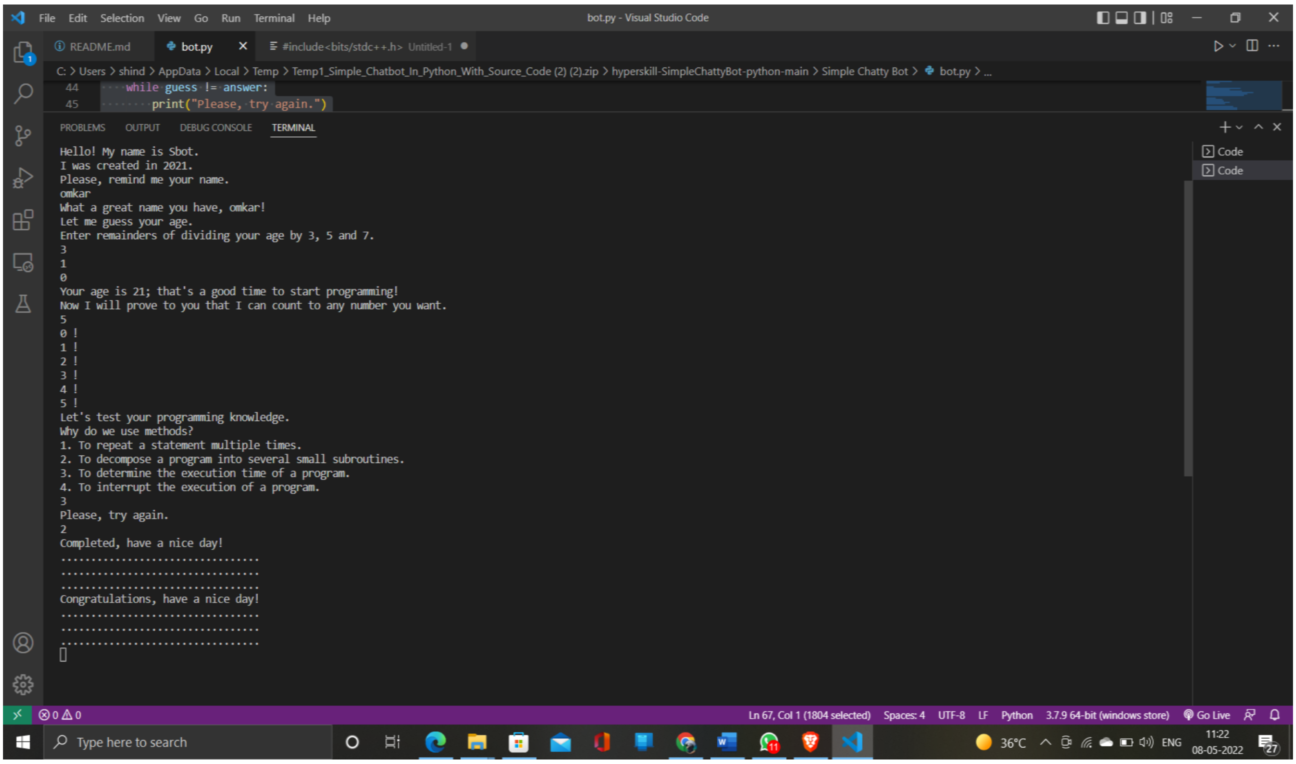
--Assignment -5 --

--Chatbot---

```
def greet(bot_name, birth_year):
    print("Hello! My name is {0}.".format(bot_name))
    print("I was created in {0}.".format(birth_year))


def remind_name():
    print('Please, remind me your name.')
    name = input()
    print("What a great name you have, {0}!".format(name))


def guess_age():
    print('Let me guess your age.')
    print('Enter remainders of dividing your age by 3, 5 and 7.')

    rem3 = int(input())
    rem5 = int(input())
    rem7 = int(input())
    age = (rem3 * 70 + rem5 * 21 + rem7 * 15) % 105

    print("Your age is {0}; that's a good time to start
programming!".format(age))


def count():
    print('Now I will prove to you that I can count to any number you want.')
    num = int(input())

    counter = 0
    while counter <= num:
        print("{0} !".format(counter))
        counter += 1


def test():
    print("Let's test your programming knowledge.")
    print("Why do we use methods?")
    print("1. To repeat a statement multiple times.")
    print("2. To decompose a program into several small subroutines.")
    print("3. To determine the execution time of a program.")
    print("4. To interrupt the execution of a program.")
    answer = 2
    guess = int(input())
    while guess != answer:
        print("Please, try again.")
        guess = int(input())

    print('Completed, have a nice day!')
    print('.................................')
    print('.................................')
    print('.................................')


def end():
    print('Congratulations, have a nice day!')
    print('.................................')
    print('.................................')
    print('.................................')
    input()

greet('Sbot', '2021')  # change it as you need
remind_name()
guess_age()
count()
test()
end()
```

```
OutputHello! My name is Sbot.

I was created in 2021.

Please, remind me your name.

omkar

What a great name you have, omkar!

Let me guess your age.

Enter remainders of dividing your age by 3, 5 and 7.

3

1

0

Your age is 21; that's a good time to start programming!
```

```
Now I will prove to you that I can count to any number you want.
5
0 !
1 !
2 !
3 !
4 !
5 !
Let's test your programming knowledge.
Why do we use methods?
1. To repeat a statement multiple times.
2. To decompose a program into several small subroutines.
3. To determine the execution time of a program.
4. To interrupt the execution of a program.
3
Please, try again.
2
Completed, have a nice day!
................................
................................
................................
Congratulations, have a nice day!
................................
................................
................................:
```


File
Edit
Selection
View
Go
Run
Terminal
Help
bot.py - Visual Studio Code
README.md
bot.py
#include<bits/stdc++.h> Untitled-1
C: > Users > shind > AppData > Local > Temp > Temp1_Simple_Chatbot_In_Python_With_Source_Code (2) (2).zip > hyperskill-SimpleChattyBot-python-main > Simple Chatty Bot > bot.py > ...
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Code
0 0
Ln 67, Col 1 (1804 selected)
Spaces: 4
UTF-8
LF
Python
3.7.9 64-bit (windows store)
Go Live
Type here to search
36°C
ENG
11:22
08-05-2022

```python
    while guess != answer:
        print("Please, try again.")
```

```
Hello! My name is Sbot.
I was created in 2021.
Please, remind me your name.
omkar
What a great name you have, omkar!
Let me guess your age.
Enter remainders of dividing your age by 3, 5 and 7.
3
1
0
Your age is 21; that's a good time to start programming!
Now I will prove to you that I can count to any number you want.
5
0 !
1 !
2 !
3 !
4 !
5 !
Let's test your programming knowledge.
Why do we use methods?
1. To repeat a statement multiple times.
2. To decompose a program into several small subroutines.
3. To determine the execution time of a program.
4. To interrupt the execution of a program.
3
Please, try again.
2
Completed, have a nice day!
.................................
.................................
.................................
Congratulations, have a nice day!
.................................
.................................
.................................
```