program for PT-109 American patrol boat, under the command of Lieutenant John F. Kennedy, was sunk by a Japanese destroyer, a message was received at an Australian wireless station in Playfair code: KXJEY UREBE ZWEHE WRYTU HEYFS KREHE GOYFI WTTTU OLKSY CAJPO BOTEI ZONTX BYBNT GONEY CUZWR GDSON SXBOU YWRHE BAAHY USEDQ 10. Write a C program for Playfair matrix: M F H I/J K U N O P Q Z V W X Y E L A R G D S T B C Encrypt this message: Must see you over Cadogan West. Coming at once.

Program:

# Playfair Cipher Decryption (Fixed Matrix, safe lookup)

matrix = [

['M', 'F', 'H', 'I', 'K'],

['U', 'N', 'O', 'P', 'Q'],

['Z', 'V', 'W', 'X', 'Y'],

['E', 'L', 'A', 'R', 'G'],

['D', 'S', 'T', 'B', 'C']

]

# Function to find position of letter in matrix

def find\_position(ch):

if ch == 'J':

ch = 'I' # I/J merged

for i in range(5):

for j in range(5):

if matrix[i][j] == ch:

return i, j

return None, None # not found

def playfair\_decrypt(ciphertext):

# Remove spaces and non-letters

import re

ciphertext = re.sub(r'[^A-Z]', '', ciphertext.upper())

plaintext = ""

# Make sure length is even

if len(ciphertext) % 2 != 0:

ciphertext += 'X'

i = 0

while i < len(ciphertext):

a, b = ciphertext[i], ciphertext[i+1]

r1, c1 = find\_position(a)

r2, c2 = find\_position(b)

if None in (r1, c1, r2, c2):

# Skip this pair if any letter not found

i += 2

continue

if r1 == r2: # same row

plaintext += matrix[r1][(c1 - 1) % 5]

plaintext += matrix[r2][(c2 - 1) % 5]

elif c1 == c2: # same column

plaintext += matrix[(r1 - 1) % 5][c1]

plaintext += matrix[(r2 - 1) % 5][c2]

else: # rectangle

plaintext += matrix[r1][c2]

plaintext += matrix[r2][c1]

i += 2

return plaintext

# PT-109 ciphertext

ciphertext = """KXJEY UREBE ZWEHE WRYTU HEYFS

KREHE GOYFI WTTTU OLKSY CAJPO

BOTEI ZONTX BYBNT GONEY CUZWR

GDSON SXBOU YWRHE BAAHY USEDQ"""

plaintext = playfair\_decrypt(ciphertext)

print("Decrypted Plaintext:\n", plaintext)output:

