1.**Write a C Program to calculate Electricity Bill**

Given an integer U denoting the amount of KWh units of electricity consumed, the task is to calculate the electricity bill with the help of the below charges:  
• 1 to 100 units – Rs. 10/- Per Unit  
• 100 to 200 units – Rs. 15/- Per Unit  
• 200 to 300 units – Rs. 20/- Per Unit  
• above 300 units – Rs. 25/- Per Unit

Examples:

Input: U = 250

Output: 3500

Explanation:

Charge for the first 100 units – 10100 = 1000Charge for the 100 to 200 units – 15100 = 1500  
Charge for the 200 to 250 units – 20\*50 = 1000  
Total Electricity Bill = 1000 + 1500 + 1000 = 3500

**Program**

#include <stdio.h>

#include <stdlib.h>

int main()

{

int unit;

printf("U=");

scanf("%d",&unit);

if(unit<=100){

printf("%d",unit\*10);

}

else if(unit<=200){

printf("%d",(100\*5)+(unit-100)\*15);

}

else if(unit<=300){

printf("%d",(100\*10)+(100\*15)+(unit-200)\*20);

}

else if(unit>300){

printf("%d",(100\*10)+(100\*15)+(100\*20)+(unit-300)\*25);

}

else{

printf("No value");

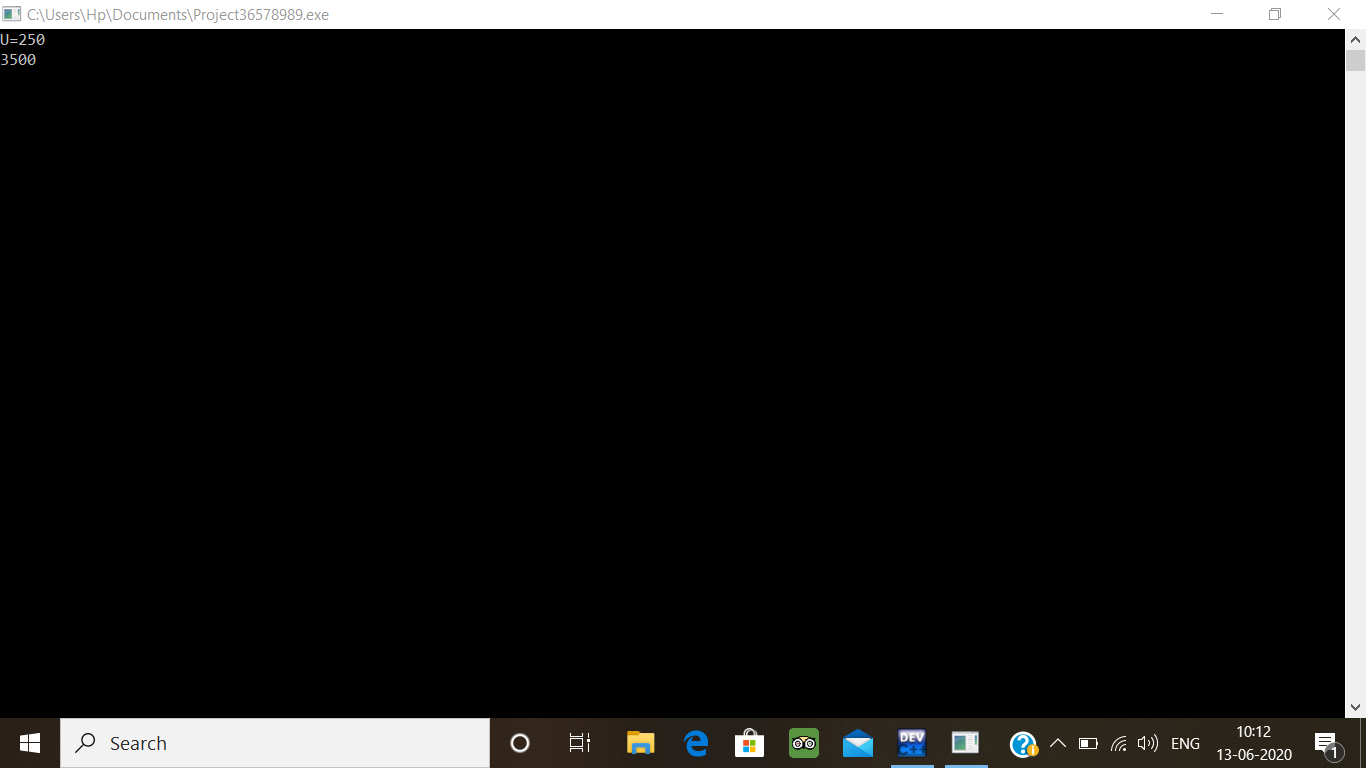
}

getch();

return 0;

}

**Output:**



**2. How to find the first non repeated character of a given String?**

import java.util.\*;

public class Main {

public static void main(String[] args) {

String str1 = "gibblegabbler";

System.out.println("The given string is: " + str1);

for (int i = 0; i < str1.length(); i++) {

boolean unique = true;

for (int j = 0; j < str1.length(); j++) {

if (i != j && str1.charAt(i) == str1.charAt(j)) {

unique = false;

break;

}

}

if (unique) {

System.out.println("The first non repeated character in String is: " + str1.charAt(i));

break;

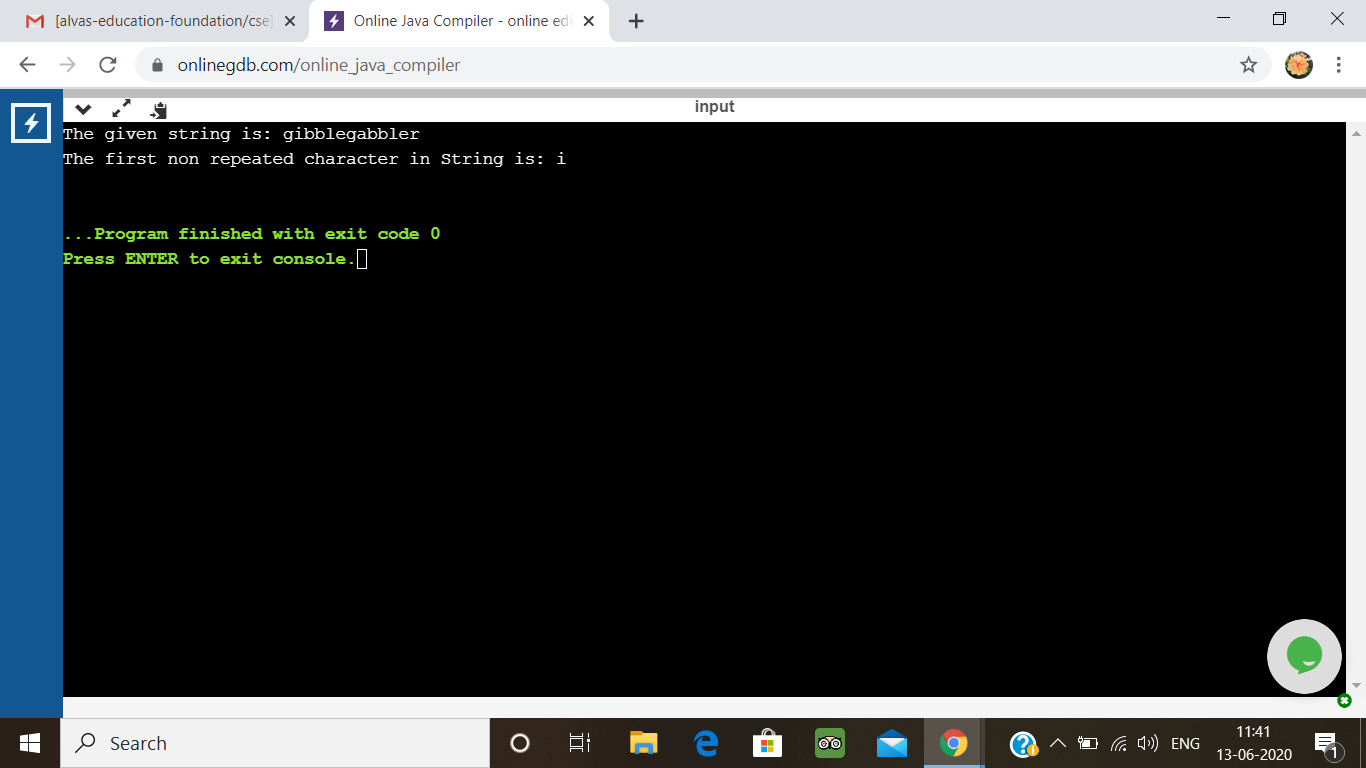
}

}

}

}

**OUTPUT**



**3. Write a Java Program to determine whether a given matrix is a sparse matrix**

Description:  
Algorithm  
STEP 1: START  
STEP 2: DEFINE rows, cols, size  
STEP 3: SET count = 0  
STEP 4: INITIALIZE first matrix a[][] ={{4,0,0}, {0,5,0}, {0,0,6}}  
STEP 5: rows = a.length  
STEP 6: cols = a[0].length  
STEP 7: size = rows\*cols  
STEP 8: REPEAT STEP 9 to STEP 10 UNTIL i<rows  
//for(i=0;i<rows; i++)  
STEP 9: REPEAT STEP 10 UNTIL j<cols  
//for(j=0;j<cols; j++)  
STEP 10: if(a[i][j]==0) then count++  
STEP 11: if(count>size/2) then PRINT "Yes" else PRINT "No"  
STEP 12: END

public class Main

{

public static void main(String[] args) {

int rows, cols, size, count = 0;

int a[][] = {

{4, 0, 0},

{0, 5, 0},

{0, 0, 6}

};

rows = a.length;

cols = a[0].length;

size = rows \* cols;

for(int i = 0; i < rows; i++){

for(int j = 0; j < cols; j++){

if(a[i][j] == 0)

count++;

}

}

if(count > (size/2))

System.out.println("Given matrix is a sparse matrix");

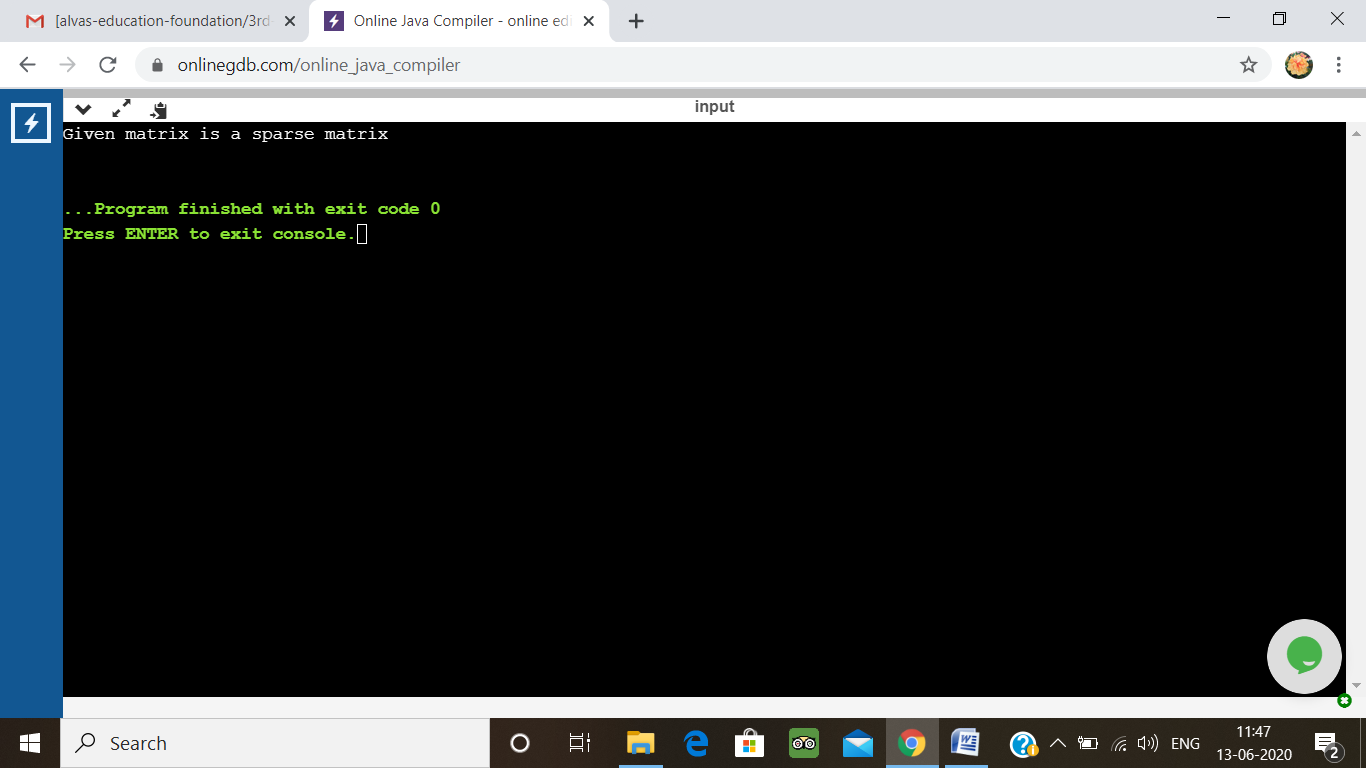
else

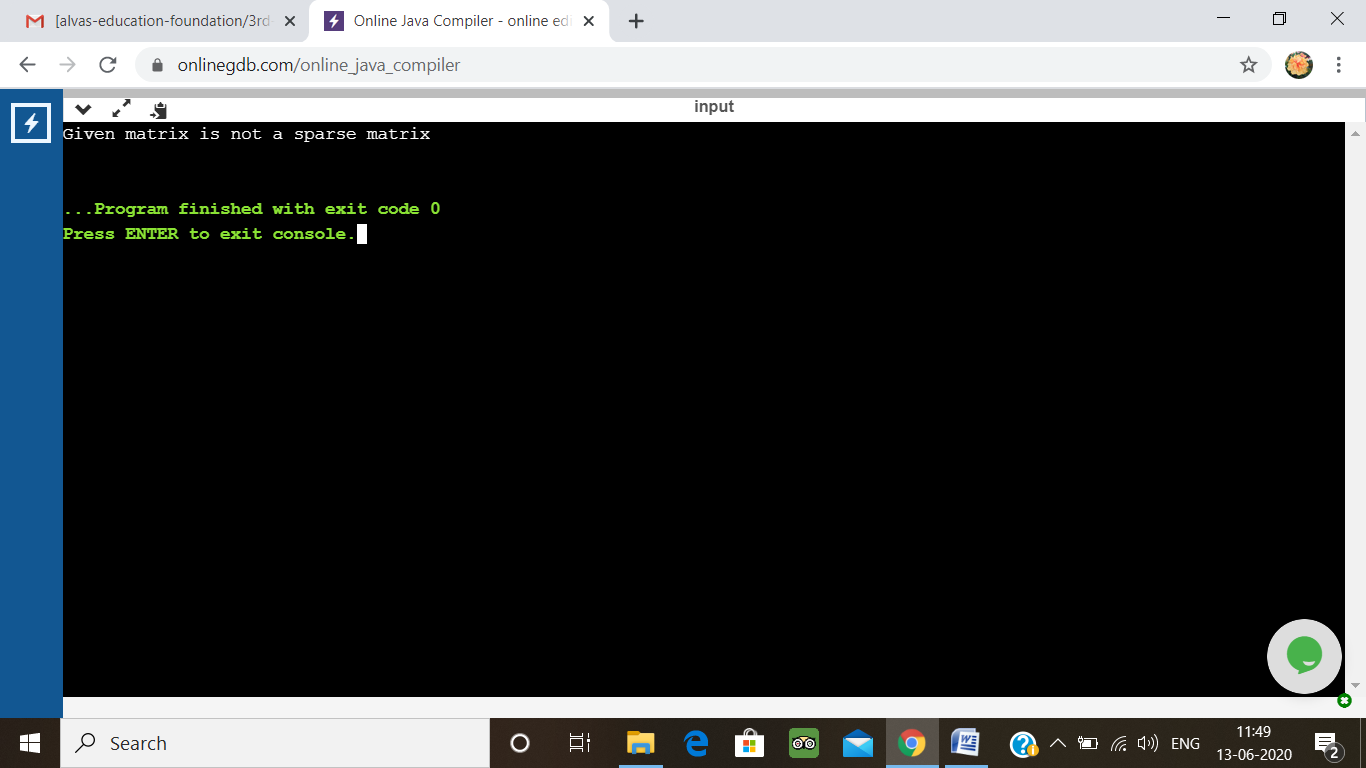
System.out.println("Given matrix is not a sparse matrix");

}

}

**OUTPUT**





**4.Python Program to print the pattern**

Description:  
Input:  
Number of rows is 5

Output Pattern is:  
A  
B C  
D E F  
G H I J  
K L M N O

def contalpha(n):

num = 65

for i in range(0, n):

for j in range(0, i+1):

ch = chr(num)

print(ch, end=" ")

num = num +1

print("\r")

n = 5

contalpha(n)

**OUTPUT**

