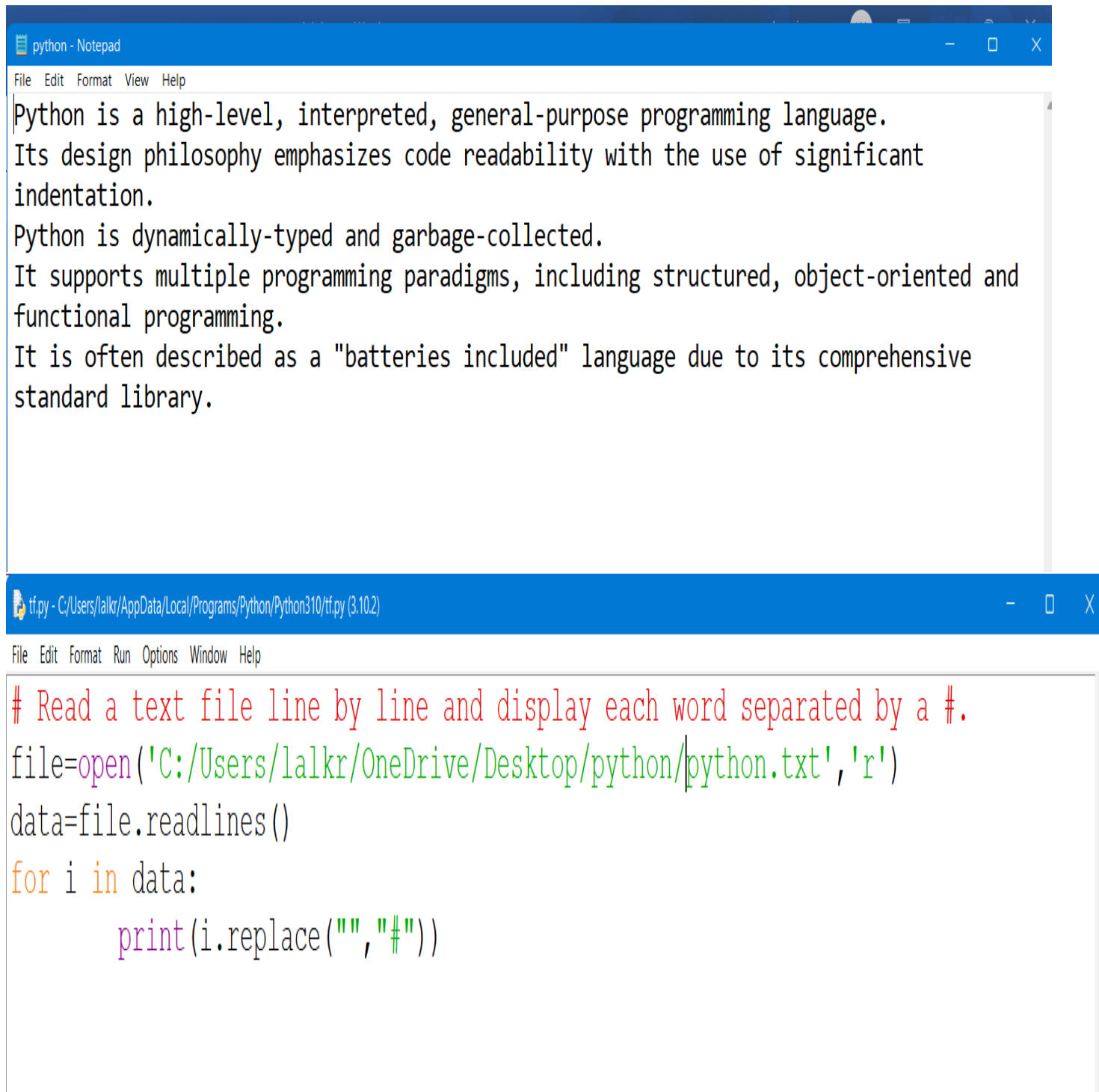


PROGRAM (1).

INPUT



The image shows two overlapping windows. The top window is a Notepad application titled 'python - Notepad'. It contains the following text:

```
Python is a high-level, interpreted, general-purpose programming language.  
Its design philosophy emphasizes code readability with the use of significant  
indentation.  
Python is dynamically-typed and garbage-collected.  
It supports multiple programming paradigms, including structured, object-oriented and  
functional programming.  
It is often described as a "batteries included" language due to its comprehensive  
standard library.
```

The bottom window is a Python IDE titled 'tf.py - C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py (3.10.2)'. It contains the following Python code:

```
# Read a text file line by line and display each word separated by a #.  
file=open('C:/Users/lalkr/OneDrive/Desktop/python/python.txt','r')  
data=file.readlines()  
for i in data:  
    print(i.replace(" ", "#"))
```

OUTPUT

```
IDLE Shell 3.10.2
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py =====
#P#y#t#h#o#n# #i#s# #a# #h#i#g#h#-#l#e#v#e#l#,# #i#n#t#e#r#p#r#e#t#e#d#,# #g#e#n#e#r#a#l#-#p#u#r#p#o#s#e# #p#r#o#g#r#a#m#m#i#n#g# #l#a#n#g#u#a#g#e#.#
#
#I#t#s# #d#e#s#i#g#n# #p#h#i#l#o#s#o#p#h#y# #e#m#p#h#a#s#i#z#e#s# #c#o#d#e# #r#e#a#d#a#b#i#l#i#t#y# #w#i#t#h# #t#h#e# #u#s#e# #o#f# #s#i#g#n#i#f#i#c#a#n#t# #i#n#d#e#n#t#a#t#i#o#n#.#
#
#P#y#t#h#o#n# #i#s# #d#y#n#a#m#i#c#a#l#l#y#-#t#y#p#e#d# #a#n#d# #g#a#r#b#a#g#e#-#c#o#l#l#e#c#t#e#d#.# #
#
#I#t# #s#u#p#p#o#r#t#s# #m#u#l#t#i#p#l#e# #p#r#o#g#r#a#m#m#i#n#g# #p#a#r#a#d#i#g#m#s#,# #i#n#c#l#u#d#i#n#g# #s#t#r#u#c#t#u#r#e#d#,# #o#b#j#e#c#t#-#o#r#i#e#n#t#e#d# #a#n#d# #f#u#n#c#t#i#o#n#a#l# #p#r#o#g#r#a#m#m#i#n#g#.#
#
#I#t# #i#s# #o#f#t#e#n# #d#e#s#c#r#i#b#e#d# #a#s# #a# #"#b#a#t#t#e#r#i#e#s# #i#n#c#l#u#d#e#d#"# #l#a#n#g#u#a#g#e# #d#u#e# #t#o# #i#t#s# #c#o#m#p#r#e#h#e#n#s#i#v#e# #s#t#a#n#d#a#r#d# #l#i#b#r#a#r#y#.#
>>>
```

```
Vowels are : 39
consonants are : 61
Lower case letters are : 98
Upper case letters are : 2
>>> |
```

```
# Read a text file and display the number of
# vowels/consonants/uppercase/lowercase characters in the file.
f=open("test2.txt","r")
cont=f.read()
v=0
cons=0
l_c_l=0
u_c_l=0
for ch in cont:
    if (ch.islower()):
        l_c_l+=1
    elif(ch.isupper()):
        u_c_l+=1
    ch=ch.lower()
    if (ch in ['a','e','i','o','u']):
        v+=1
    elif (ch in ['b','c','d','f','g',
                'h','j','k','l','m',
                'n','p','q','r','s',
                't','v','w','x','y','z']):
        cons+=1
f.close()
print("Vowels are : ",v)
print("consonants are : ",cons)
print("Lower case letters are : ",l_c_l)
print("Upper case letters are : ",u_c_l)
```

OUTPUT OF PROGRAM (2).

INPUT OF PROGRAM (3

```
#Remove all the lines that contain the character 'a' in a file and write it to another file.
print("Remove all the lines that contain the character 'a' in a file and write it to another file. ")
myfile = open("book.txt", "r")
newfile = open("story.txt", "w")
line = myfile.readlines()
print()
print("data in first file ")
print(line)
print()
for data in line:
    if 'a' not in data:
        newfile.write(data)
newfile= open("story.txt", "r")
line = newfile.readlines()
print("Data get copied in second file")
print(line)
```

OUTPUT OF PROGRAM (3).

Remove all the lines that contain the character 'a' in a file and write it to another file.

data in first file

['Write\n', 'a \n', 'Python\n', 'program\n', 'to \n', 'implement \n', 'a \n', 'stack \n', 'using \n', 'list.']

Data get copied in second file

['Write\n', 'Python\n', 'to \n', 'implement \n', 'using \n', 'list.']

>>> |

INPUT OF PROGRAM (4

```
#Create a binary file with name and roll number.
#Search for a given roll number and display the name, if not found
#display appropriate messageimport pickle
import sys
import csv
dict={}
def write_in_file():
    file=open("stud2.csv","ab")
    no=int(input("ENTER NO OF STUDENTS: "))
    for i in range(no):
        print("Enter details of student ", i+1)
        dict["roll"]=int(input("Enter roll number: "))
        dict["name"]=input("enter the name: ")
        pickle.dump(dict,file)
    file.close()
def display():
    file=open("stud2.csv","rb")
    try:
        while True:
            stud=pickle.load(file)
            print(stud)
    except EOFError:
        pass
    file.close()
def search():
    file=open("stud2.csv","rb")
    r=int(input("enter the rollno to search: "))
    found=0
    try:
        while True:
            data=pickle.load(file)
            if data["roll"]==r:
```



```

file.close()
def search():
    file=open("stud2.csv","rb")
    r=int(input("enter the rollno to search: "))
    found=0
    try:
        while True:
            data=pickle.load(file)
            if data["roll"]==r:
                print("The rollno =",r," record found")
                print(data)
                found=1
                break
    except EOFError:
        pass
    if found==0:
        print("The rollno =",r," record is not found")
    file.close()
while True:
    print("MENU \n 1-Write in a file \n 2-display ")
    print(" 3-search\n 4-exit \n")
    ch=int(input("Enter your choice = "))
    if ch==1:
        write_in_file()
    if ch==2:
        display()
    if ch==3:
        search()
    if ch==4:
        print(" Thank you ")
        sys.exit()

```

MENU

- 1-Write in a file
- 2-display
- 3-search
- 4-exit

Enter your choice = 1

ENTER NO OF STUDENTS: 2

Enter details of student 1

Enter roll number: 2

enter the name: abhigyan

Enter details of student 2

Enter roll number: 4

enter the name: anurag

MENU

- 1-Write in a file
- 2-display
- 3-search
- 4-exit

Enter your choice = 2

{'roll': 2, 'name': 'abhigyan'}

{'roll': 4, 'name': 'anurag'}

MENU

- 1-Write in a file
- 2-display
- 3-search
- 4-exit

Enter your choice = 3

enter the rollno to search: 2

The rollno = 2 record found

{'roll': 2, 'name': 'abhigyan'}

MENU

- 1-Write in a file
- 2-display
- 3-search
- 4-exit

Enter your choice = 4

Thank you

>>>|

OUTPUT OF PROGRAM (4).

INPUT OF PROGRAM (5).

#Create a binary file with roll number, name and marks.

#Input a roll number and update the marks.

```
import pickle
f=open("records.dat", "wb")
pickle.dump([1, "Wakil", 90], f)
pickle.dump([2, "Tanish", 80], f)
pickle.dump([3, "Priyashi", 90], f)
pickle.dump([4, "Kanupriya", 80], f)
pickle.dump([5, "Ashutosh", 85], f)
f.close()
f=open("records.dat", "rb")
roll=int(input("Enter the Roll Number: "))
marks=float(input("Enter the updated marks: "))
List = [ ]
flag = False
while True:
    try:
        record=pickle.load(f)
        List.append(record)
    except EOFError:
        break
f.close()
f=open("records.dat", "wb")
for rec in List:
    if rec[0]==roll:
        rec[2] = marks
        pickle.dump(rec, f)
        print("Record updated successfully")
        flag = True
    else:
```


Enter the Roll Number: 2
Enter the updated marks: 23
Record updated successfully
>>>

```
f.close()
if flag==False:
    print("This roll number does not exist")
```

OUTPUT OF PROGRAM (5).

INPUT OF PROGRAM (6).

```
# Write a random number generator that generates random numbers
# between 1 and 6 (simulates a dice).
import random
def rolladice():
    counter = 0
    myList = []
    while (counter) < 6:
        randomNumber = random.randint(1,6)
        myList.append(randomNumber)
        counter = counter + 1
    if (counter)>=6:
        pass
    else:
        return myList

# Take user input here
n=1
while(n==1):
    n = int(input("Enter 1 to roll a dice and get a random number:"))
    print(rolladice())
```

```
Enter 1 to roll a dice and get a random number:1
[2]
Enter 1 to roll a dice and get a random number:2
[6]
>>>|
```

OUTPUT OF PROGRAM (6).

INPUT OF PROGRAM (7).

```
# Write a Python program to implement a stack using list.
class Node:
    def __init__(self, data):
        self.data = data
        self.next = None

class Stack:
    def __init__(self):
        self.head = None

    def push(self, data):
        if self.head is None:
            self.head = Node(data)
        else:
            new_node = Node(data)
            new_node.next = self.head
            self.head = new_node

    def pop(self):
        if self.head is None:
            return None
        else:
            popped = self.head.data
            self.head = self.head.next
            return popped
```



```
a_stack = Stack()
while True:
    print('push <value>')
    print('pop')
    print('quit')
    do = input('What would you like to do? ').split()

    operation = do[0].strip().lower()
    if operation == 'push':
        a_stack.push(int(do[1]))
    elif operation == 'pop':
        popped = a_stack.pop()
        if popped is None:
            print('Stack is empty.')
        else:
            print('Popped value: ', int(popped))
    elif operation == 'quit':
        break
```

OUTPUT OF PROGRAM (7).

```
push <value>
pop
quit
What would you like to do? push 15
push <value>
pop
quit
What would you like to do? push 3
push <value>
pop
quit
What would you like to do? pop
Popped value: 3
push <value>
pop
quit
What would you like to do? pop
Popped value: 15
push <value>
pop
quit
What would you like to do? quit
>>> /
```

INPUT OF PROGRAM (8).

```
enter id: abhigyankushwaha72@gmail.com
enter password: abhigyan111222333888777555
press Y/y to continue and N/n to terminate the program
n
enter the user id to be searched
abhigyankushwaha72@gmail.com
abhigyan111222333888777555
>>>
```



```

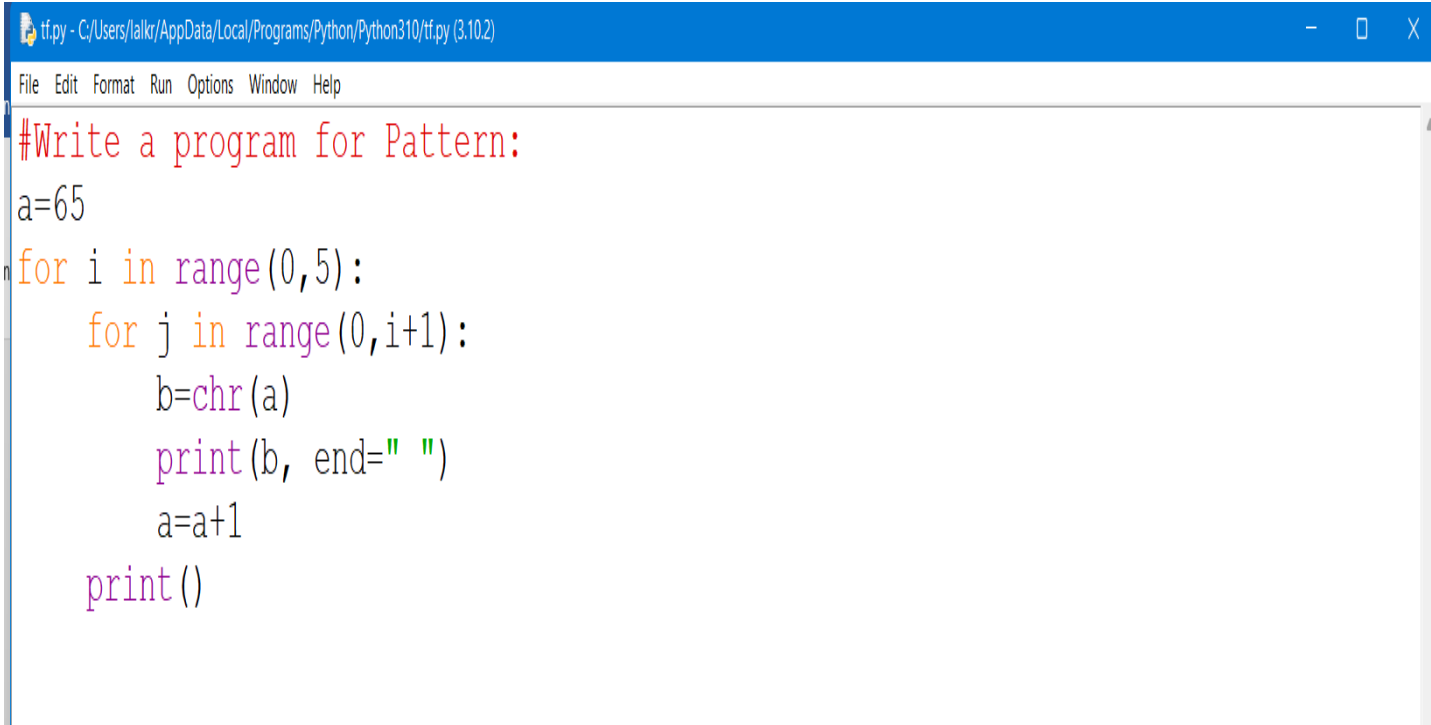
#Create a CSV file by entering user-id and password, read and
#search the password for given user id
import csv
with open("user_info.csv", "w") as obj:
    fileobj = csv.writer(obj)
    fileobj.writerow(["User Id", "password"])
    while(True):
        user_id = input("enter id: ")
        password = input("enter password: ")
        record = [user_id, password]
        fileobj.writerow(record)
        x = input("press Y/y to continue and N/n to terminate the program\n")
        if x in "Nn":
            break
        elif x in "Yy":
            continue
with open("user_info.csv", "r") as obj2:
    fileobj2 = csv.reader(obj2)
    given = input("enter the user id to be searched\n")
    for i in fileobj2:
        next(fileobj2)
        # print(i,given)
        if i[0] == given:
            print(i[1])
            break

```

OUTPUT OF PROGRAM (8).

PROGRAM 9(A).

INPUT



The screenshot shows a Python IDE window with the title bar 'tf.py - C:/Users/ialkr/AppData/Local/Programs/Python/Python310/tf.py (3.10.2)'. The menu bar includes 'File', 'Edit', 'Format', 'Run', 'Options', 'Window', and 'Help'. The code editor contains the following Python code:

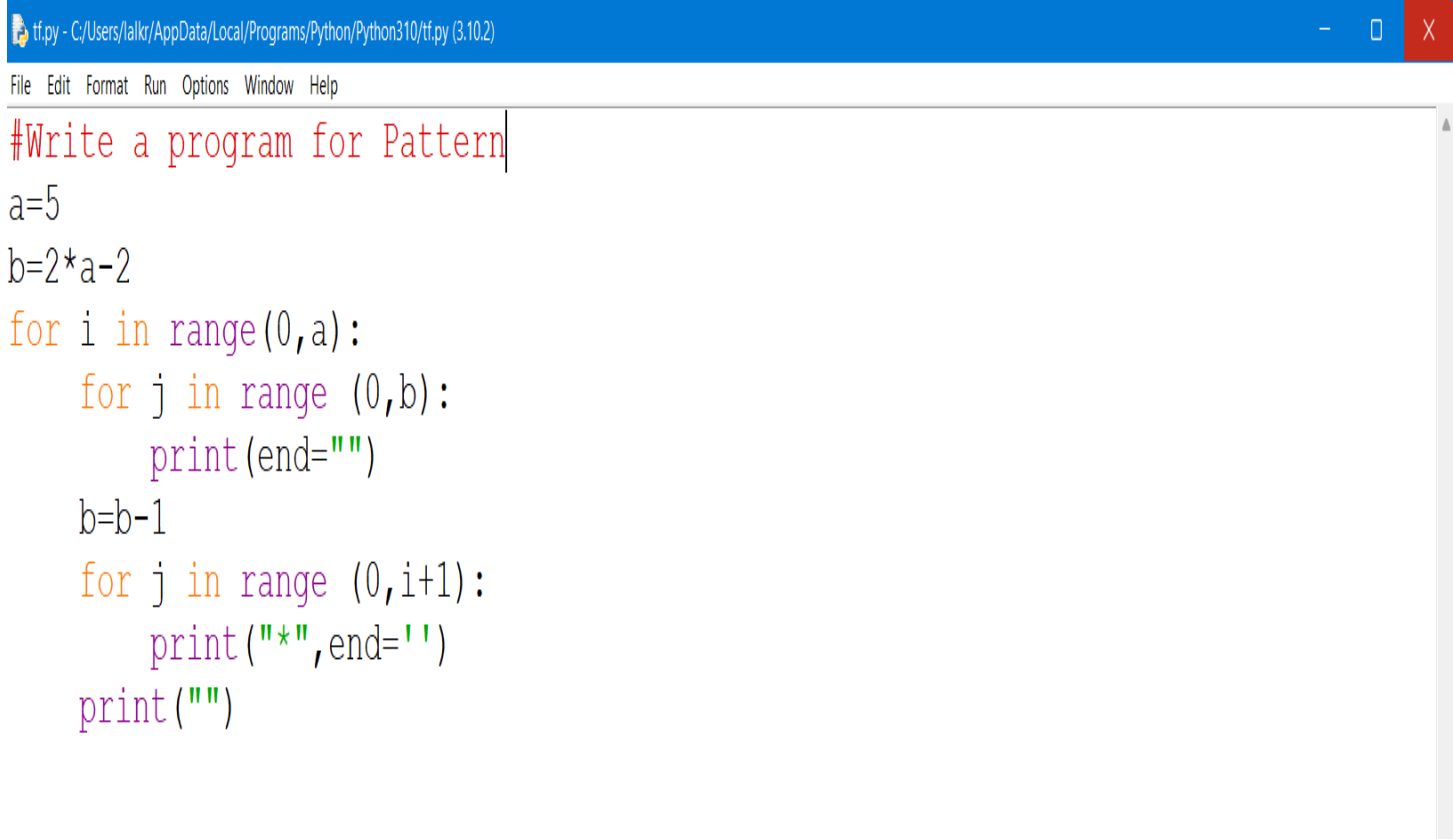
```
#Write a program for Pattern:
a=65
for i in range(0,5):
    for j in range(0,i+1):
        b=chr(a)
        print(b, end=" ")
        a=a+1
    print()
```

OUTPUT

```
IDLE Shell 3.10.2
File Edit Shell Debug Options Window Help
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py =====
A
B C
D E F
G H I J
K L M N O
>>>|
```

PROGRAM 9(B).

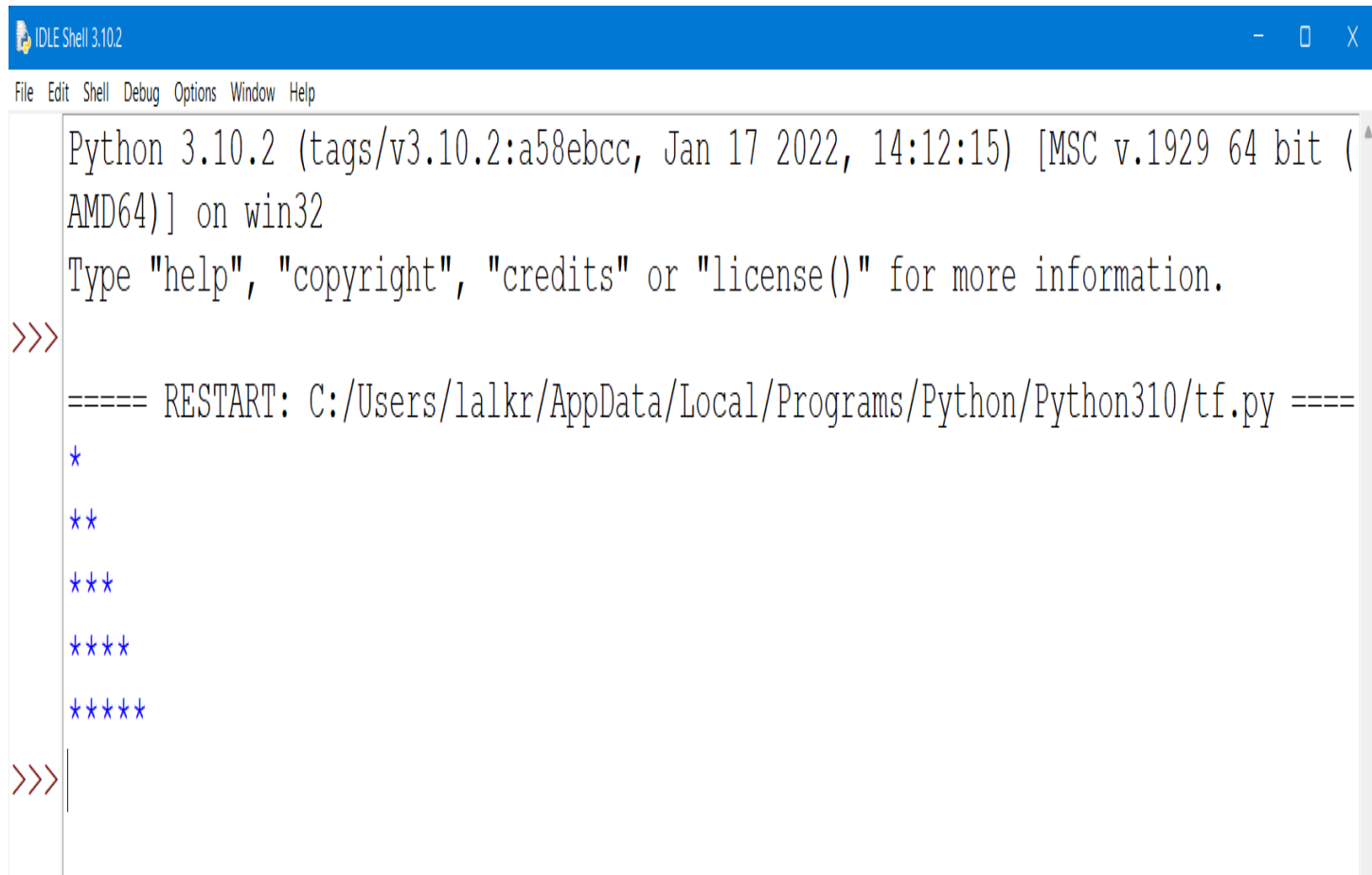
INPUT



The screenshot shows a Python IDE window titled 'tf.py - C:/Users/ialkr/AppData/Local/Programs/Python/Python310/tf.py (3.10.2)'. The menu bar includes File, Edit, Format, Run, Options, Window, and Help. The code in the editor is as follows:

```
#Write a program for Pattern
a=5
b=2*a-2
for i in range(0,a):
    for j in range (0,b):
        print(end=" ")
    b=b-1
    for j in range (0,i+1):
        print("*",end='')
    print("")
```

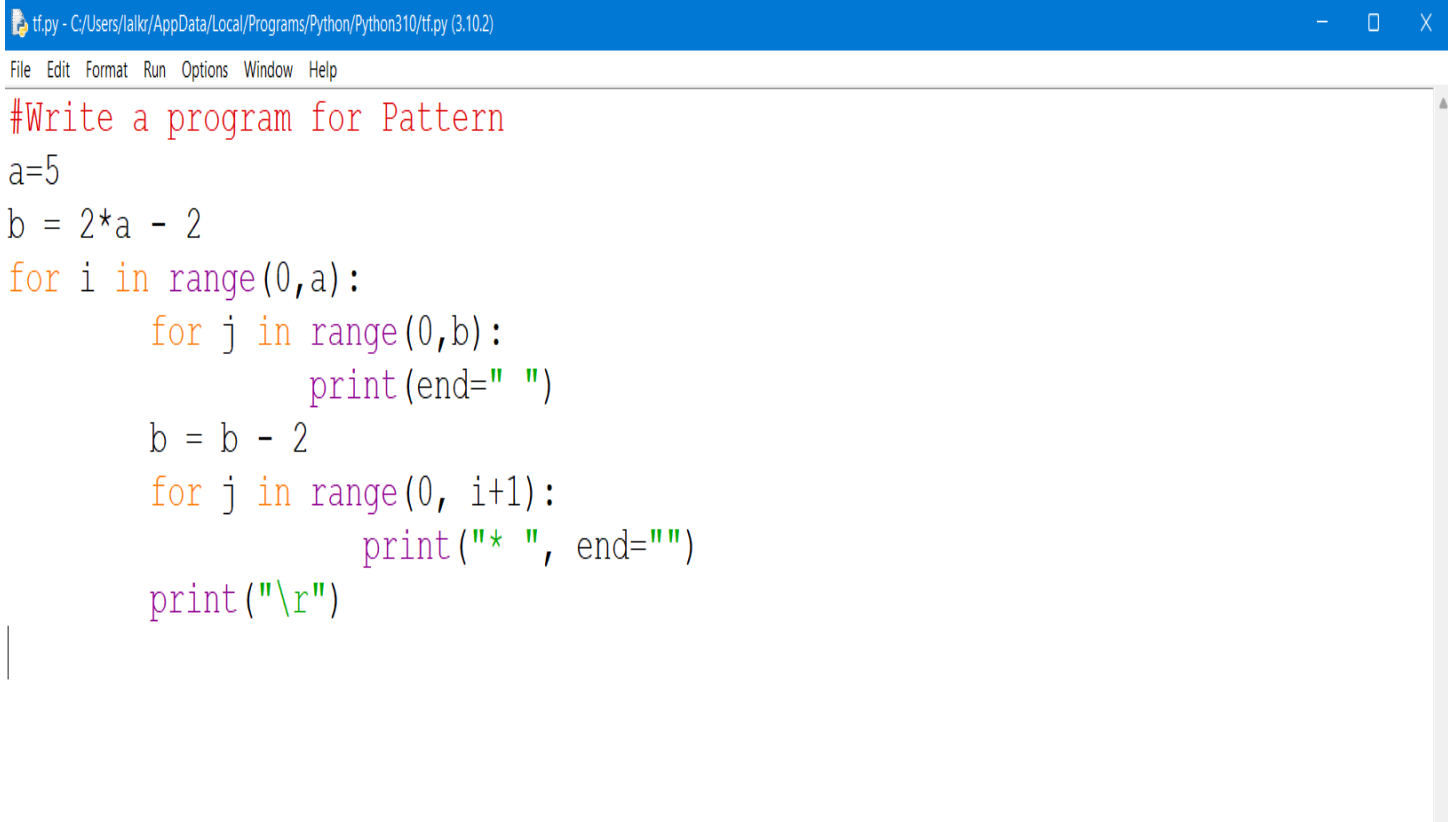

OUTPUT



```
IDLE Shell 3.10.2
File Edit Shell Debug Options Window Help
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py =====
*
**
***
****
*****
>>>|
```

PROGRAM 9(C).

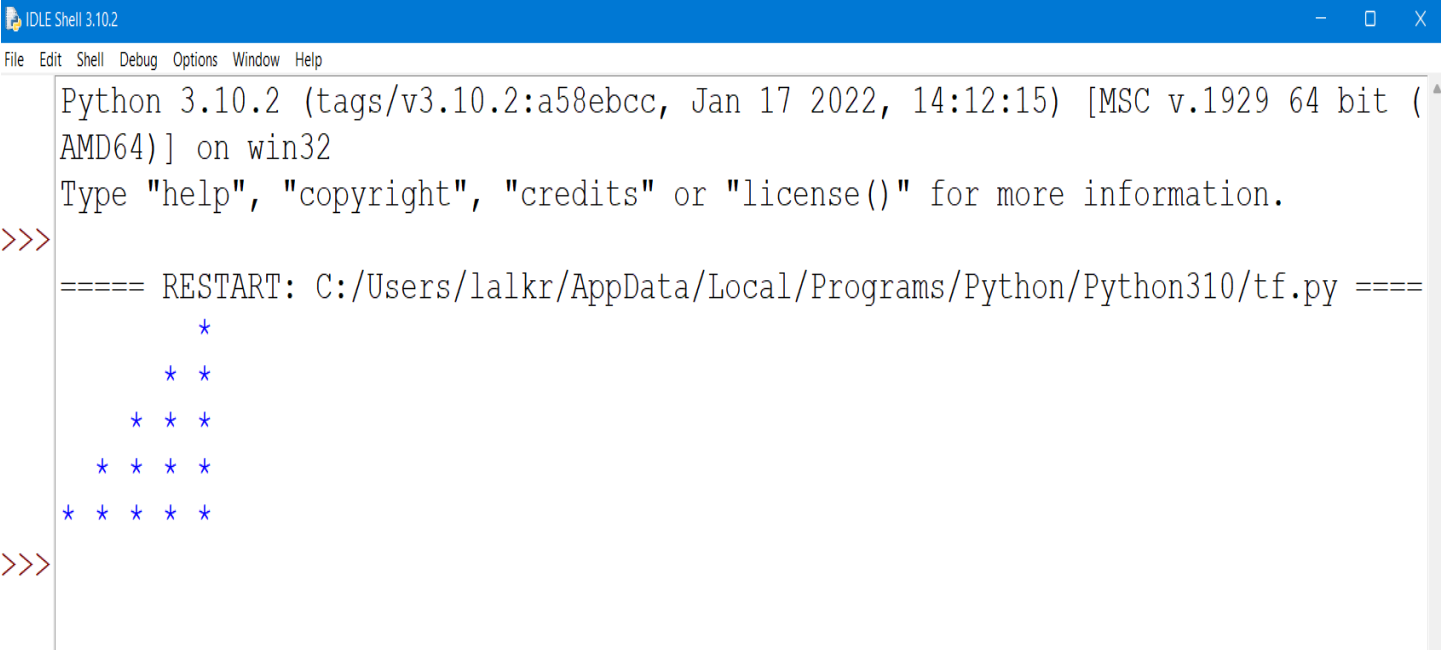
INPUT



The screenshot shows a Python IDE window with the title bar 'tf.py - C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py (3.10.2)'. The menu bar includes 'File', 'Edit', 'Format', 'Run', 'Options', 'Window', and 'Help'. The code editor contains the following Python code:

```
#Write a program for Pattern
a=5
b = 2*a - 2
for i in range(0,a):
    for j in range(0,b):
        print(end=" ")
    b = b - 2
    for j in range(0, i+1):
        print("* ", end="")
    print("\r")
```

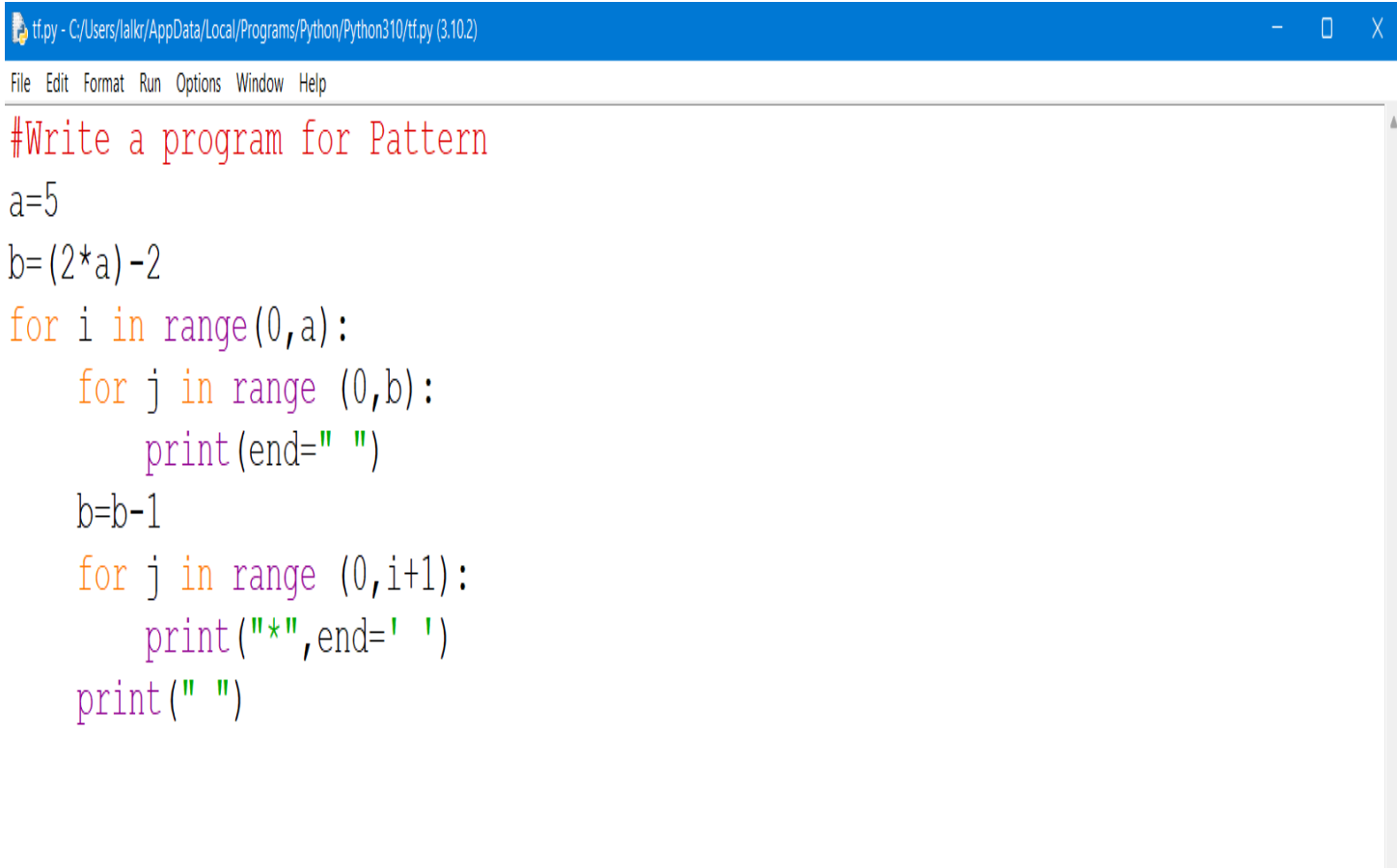
OUTPUT



```
IDLE Shell 3.10.2
File Edit Shell Debug Options Window Help
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py =====
      *
     * *
    * * *
   * * * *
  * * * * *
 * * * * *
>>>
```

PROGRAM 9(D).

INPUT



The screenshot shows a Python IDE window titled 'tf.py - C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py (3.10.2)'. The menu bar includes 'File', 'Edit', 'Format', 'Run', 'Options', 'Window', and 'Help'. The code in the editor is as follows:

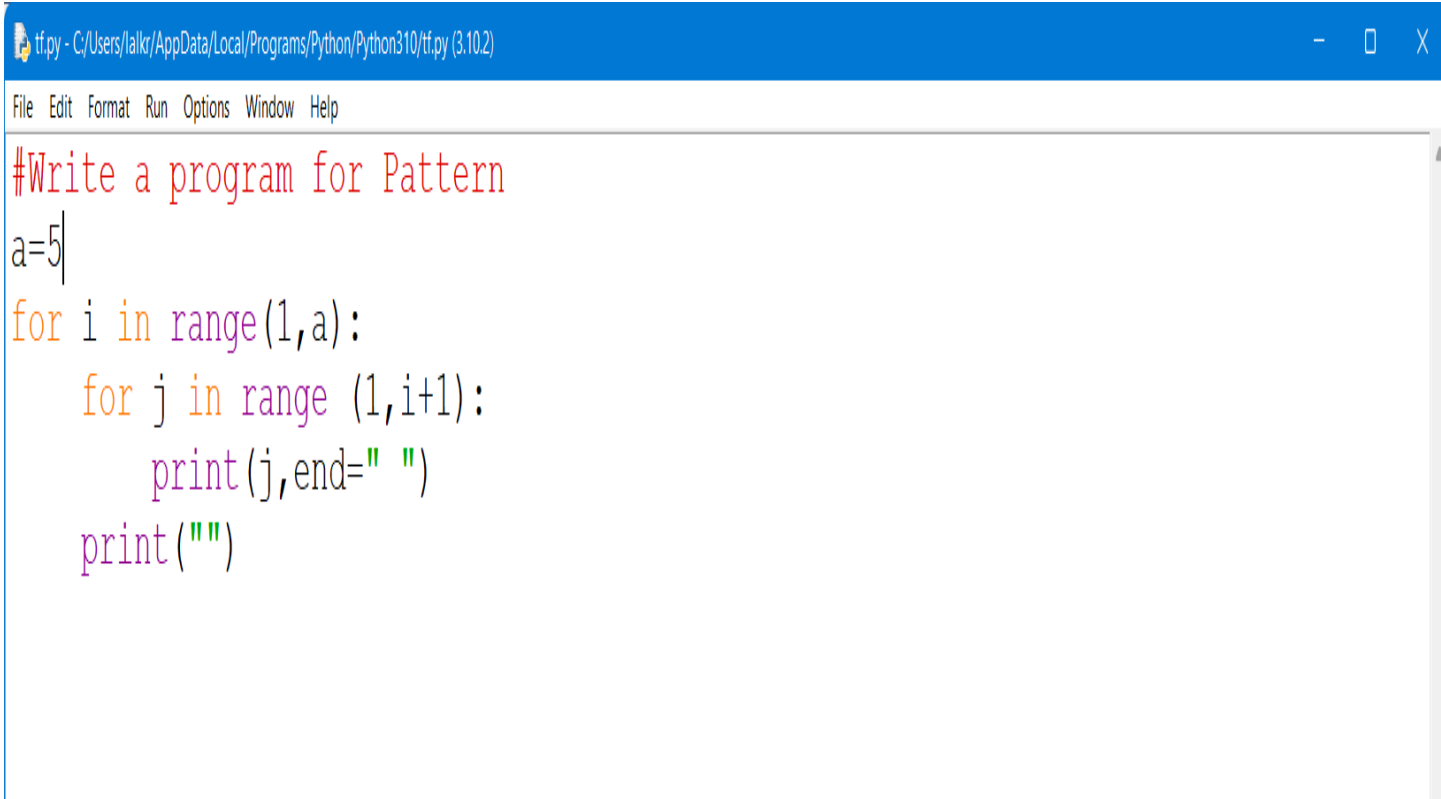
```
#Write a program for Pattern
a=5
b=(2*a)-2
for i in range(0,a):
    for j in range (0,b):
        print(end=" ")
    b=b-1
    for j in range (0,i+1):
        print("*",end=' ')
    print(" ")
```

OUTPUT

```
IDLE Shell 3.10.2
File Edit Shell Debug Options Window Help
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.192
9 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more informatio
n.
>>>
===== RESTART: C:/Users/lalkr/AppData/Local/Programs/Python/Python310/
tf.py =====
      *
    * *
  * * *
* * * *
* * * * *
>>>
```


PROGRAM 9(E).

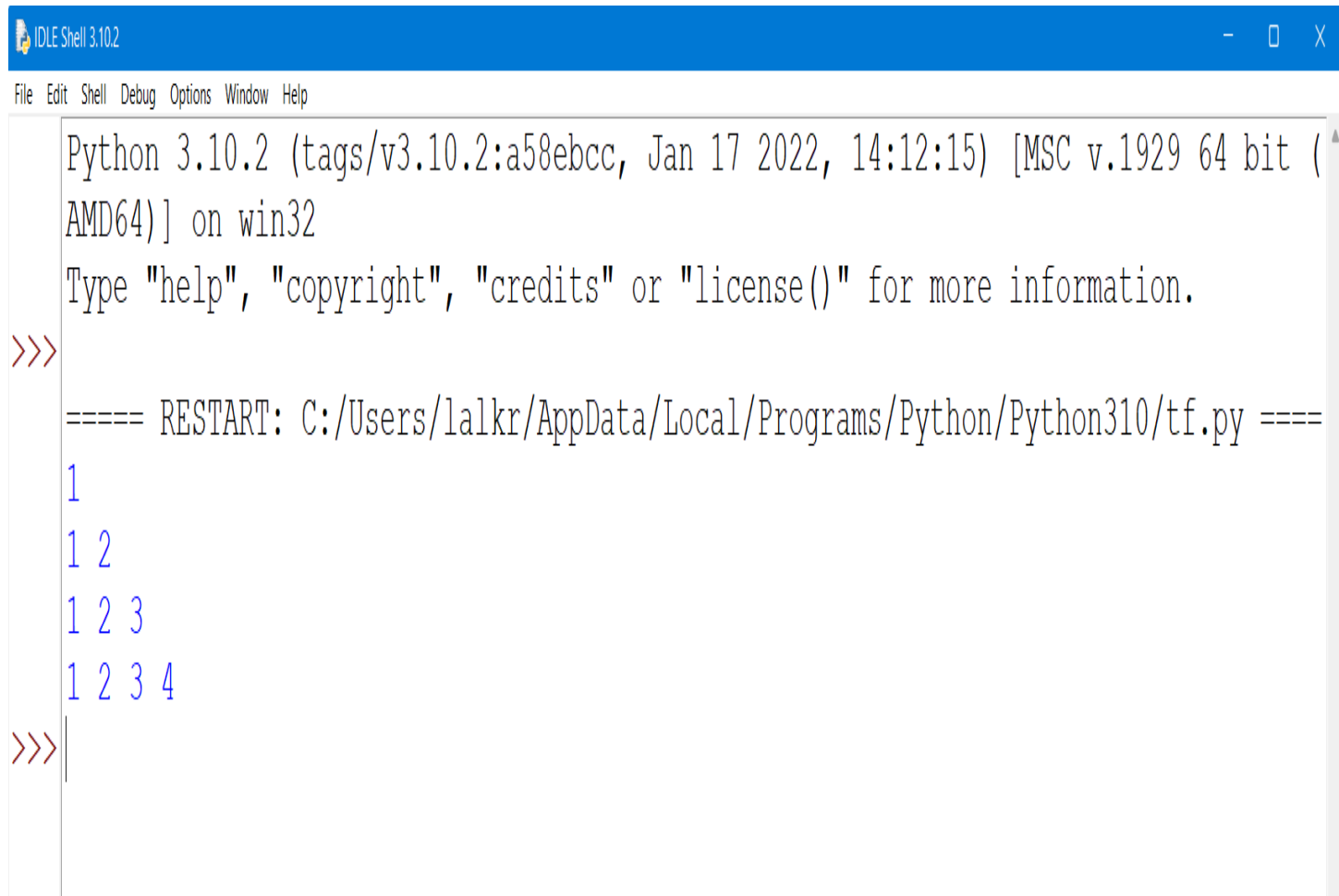
INPUT



The screenshot shows a Python IDE window with a blue title bar. The title bar text is "tf.py - C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py (3.10.2)". Below the title bar is a menu bar with "File", "Edit", "Format", "Run", "Options", "Window", and "Help". The main text area contains the following Python code:

```
#Write a program for Pattern
a=5
for i in range(1,a):
    for j in range (1,i+1):
        print(j,end=" ")
    print("")
```


OUTPUT



```
IDLE Shell 3.10.2
File Edit Shell Debug Options Window Help
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py =====
1
1 2
1 2 3
1 2 3 4
>>>|
```

PROGRAM 9(F).

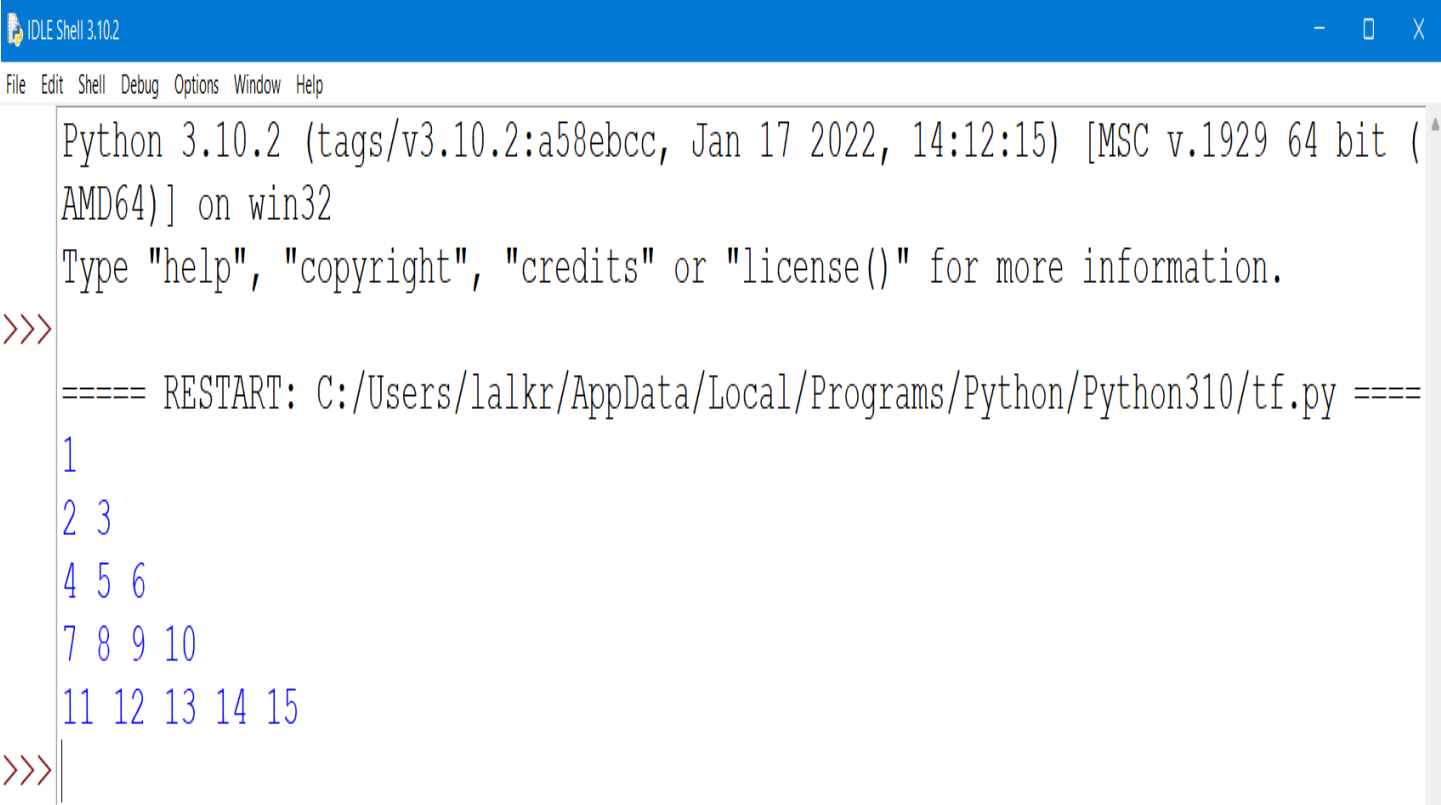
INPUT



The image shows a screenshot of a Python IDE window. The title bar at the top reads "tf.py - C:/Users/lalr/AppData/Local/Programs/Python/Python310/tf.py (3.10.2)". Below the title bar is a menu bar with the options "File", "Edit", "Format", "Run", "Options", "Window", and "Help". The main text area contains the following Python code, which is designed to print a pattern of numbers. The code uses nested loops to generate a series of numbers from 1 to 5, with each row containing one more number than the previous row. The numbers are printed with a space between them, and a new line is started after each row.

```
#Write a program for Pattern
a=1
for i in range(0,5):
    for j in range(0,i+1):
        b=a
        print(b, end=" ")
        a=a+1
    print()
```

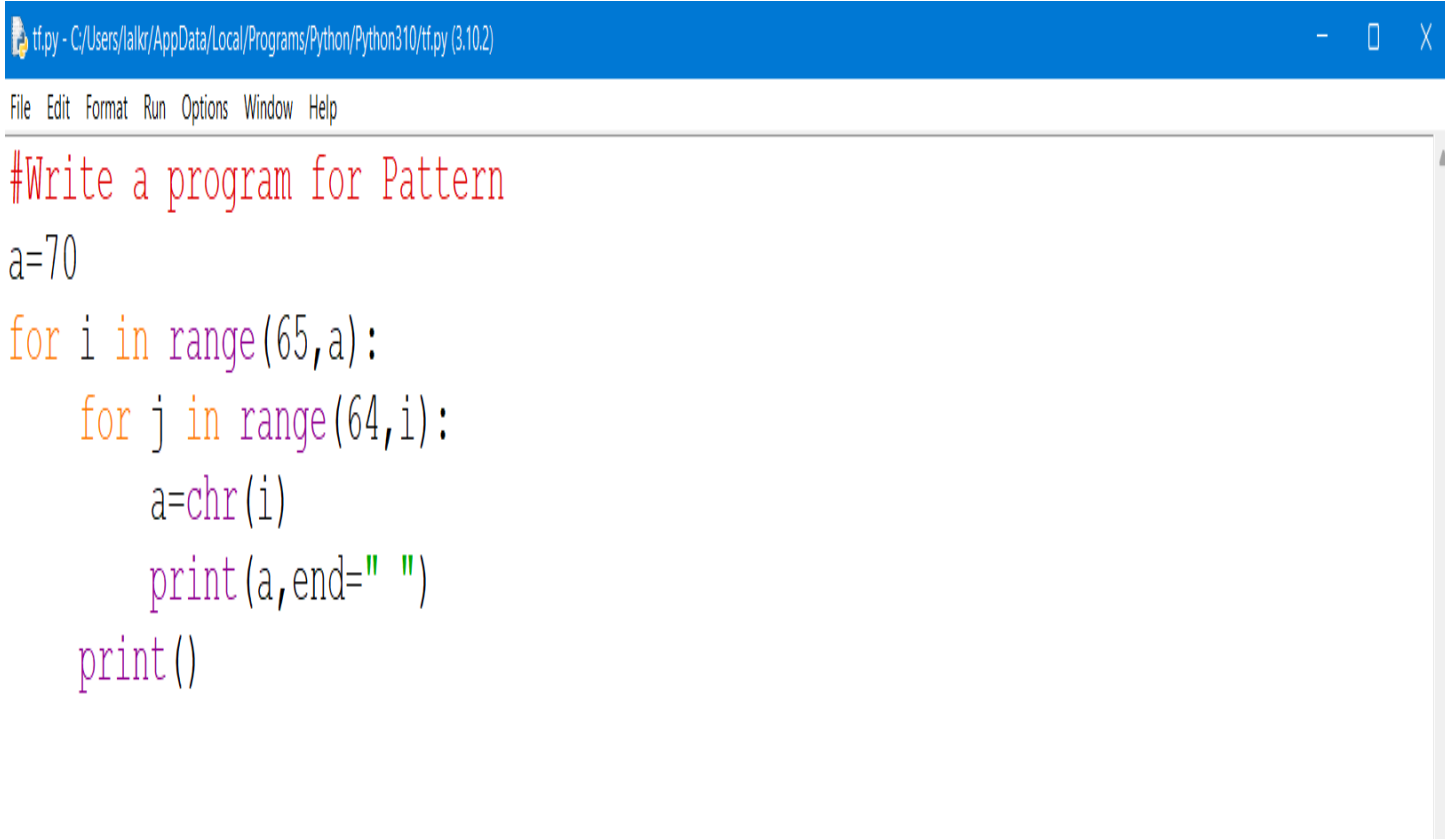
OUTPUT



```
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py =====
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
>>>|
```

PROGRAM 9(G).

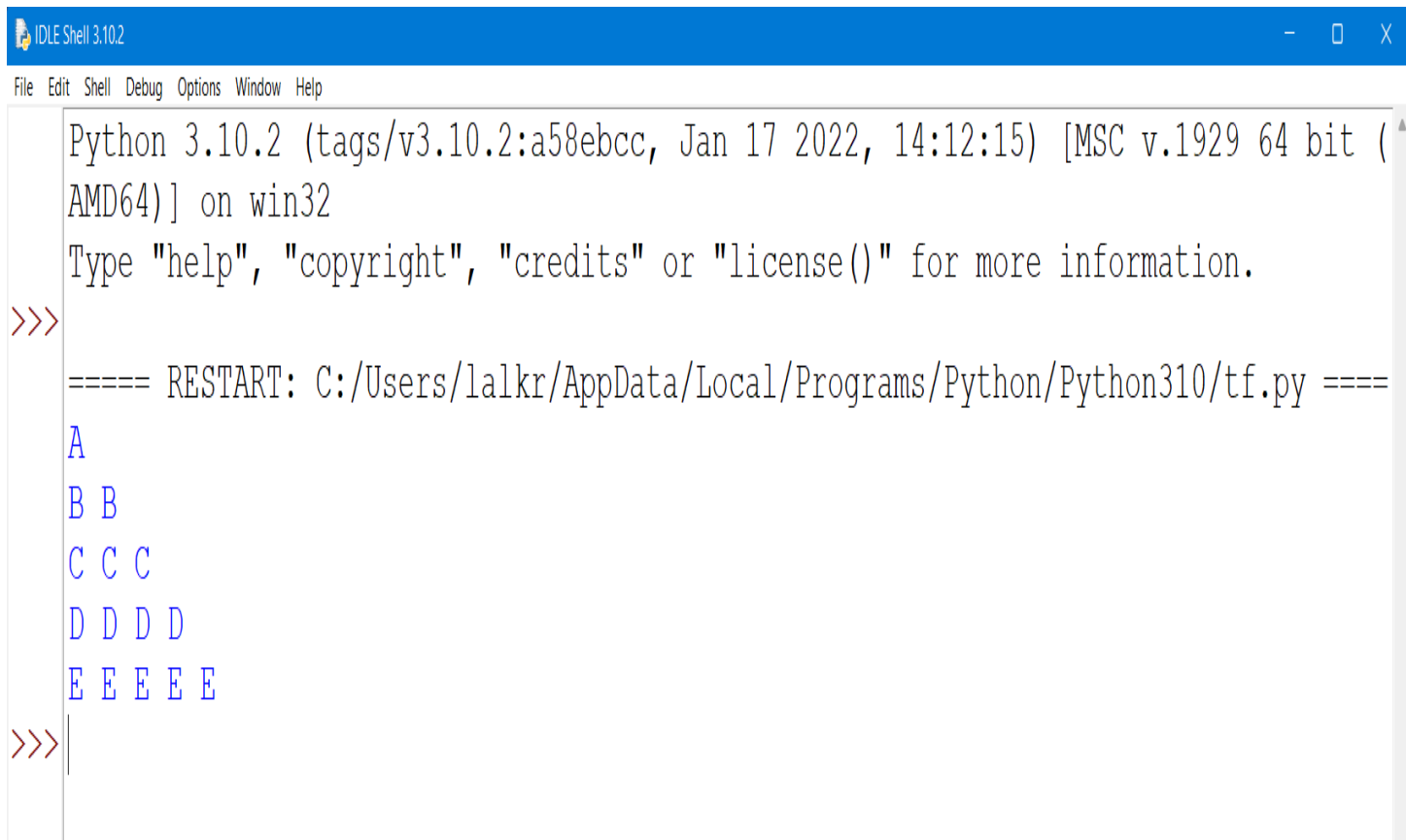
INPUT



The image shows a screenshot of a Python IDE window. The title bar at the top is blue and contains the text "tf.py - C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py (3.10.2)". Below the title bar is a menu bar with the options "File", "Edit", "Format", "Run", "Options", "Window", and "Help". The main editing area has a white background and contains the following Python code written in a monospaced font with syntax highlighting: a comment line in red, a variable assignment, and two nested for loops with indented statements. A vertical scrollbar is visible on the right side of the code editor.

```
tf.py - C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py (3.10.2)
File Edit Format Run Options Window Help
#Write a program for Pattern
a=70
for i in range(65,a):
    for j in range(64,i):
        a=chr(i)
        print(a,end=" ")
    print()
```

OUTPUT



The screenshot shows a Python IDLE Shell window. The title bar is blue and says 'IDLE Shell 3.10.2'. Below it is a menu bar with 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The main text area is white and contains the following text: 'Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on win32'. This is followed by a prompt 'Type "help", "copyright", "credits" or "license()" for more information.' and a red '>>>>' prompt. Below this is a line of text '==== RESTART: C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py =====' followed by five lines of blue text: 'A', 'B B', 'C C C', 'D D D D', and 'E E E E E'. Finally, there is another red '>>>>' prompt with a vertical cursor line next to it.

```
IDLE Shell 3.10.2
File Edit Shell Debug Options Window Help
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==== RESTART: C:/Users/lalkr/AppData/Local/Programs/Python/Python310/tf.py =====
A
B B
C C C
D D D D
E E E E E
>>>|
```