**KATHMANDU UNIVERSITY**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



**Lab Work 4**

**COMP 316**

**SUBMITTED BY: SUBMITTED TO:**

Arun Regmi Sushil Nepal

Roll No - 66 DOCSE

Computer Science (3rd Year/ 1st Sem)

Date of Submission: March 10, 2020.

# 

**Problem**

**Q.** Write a program to implement PDA.

**Key Features**

At first the language which is to be implemented for PDA is kept in a text file “input.txt” which is first accessed by a file “stack.py” which creates a stack structure for the given grammar. The program “lab4.py” compiles a given grammar structure given by the user in the program and produces a PDA. The compiled grammar is first checked to find out whether the grammar structure is applicable to produce PDA

**Code:**

**stack.py**

class Stack:

def \_init\_\_(self, size):

self.head = -1

self.stack = []

self.size = size

def isEmpty(self):

if self.head <= -1:

return True

else:

return False

def isFull(self):

if self.head >= self.size:

return True

else:

return False

def push(self, value):

if self.isFull():

pass

else:

self.head += 1

# print(self.head)

self.stack.insert(self.head, value)

def pop(self):

if self.isEmpty():

self.head -= 1

try:

self.stack.pop()

except:

pass

else:

popped = self.stack[self.head]

self.head -= 1

self.stack.pop()

return popped

def showStack(self):

if self.isEmpty():

print("Err..Stack is Empty")

else:

print(' '.join(self.stack))

**lab4.py**

#Write a program to implement PDA.

from stack import Stack

print("PDA Language")

print("---------------------------------------------------------------------")

allInputs = [line.rstrip() for line in open("input.txt")]

print(allInputs)

for inString in allInputs:

print("Give String is : {inString}")

print("---------------------------------------------------------------------")

mystack = Stack(len(inString))

print("In Values Is Equal My Stack")

print("---------------------------------------------------------------------")

for alpha in inString:

if alpha == 'a':

mystack.push(alpha)

else:

mystack.pop()

print(f" {alpha} {mystack.isEmpty()} {' '.join(mystack.stack)}")

print("---------------------------------------------------------------------")

if mystack.isEmpty():

print("a's and b's are Equal")

else :

print("a's and b's are Not Equal")

print("================================================================

====")

**editor.txt**

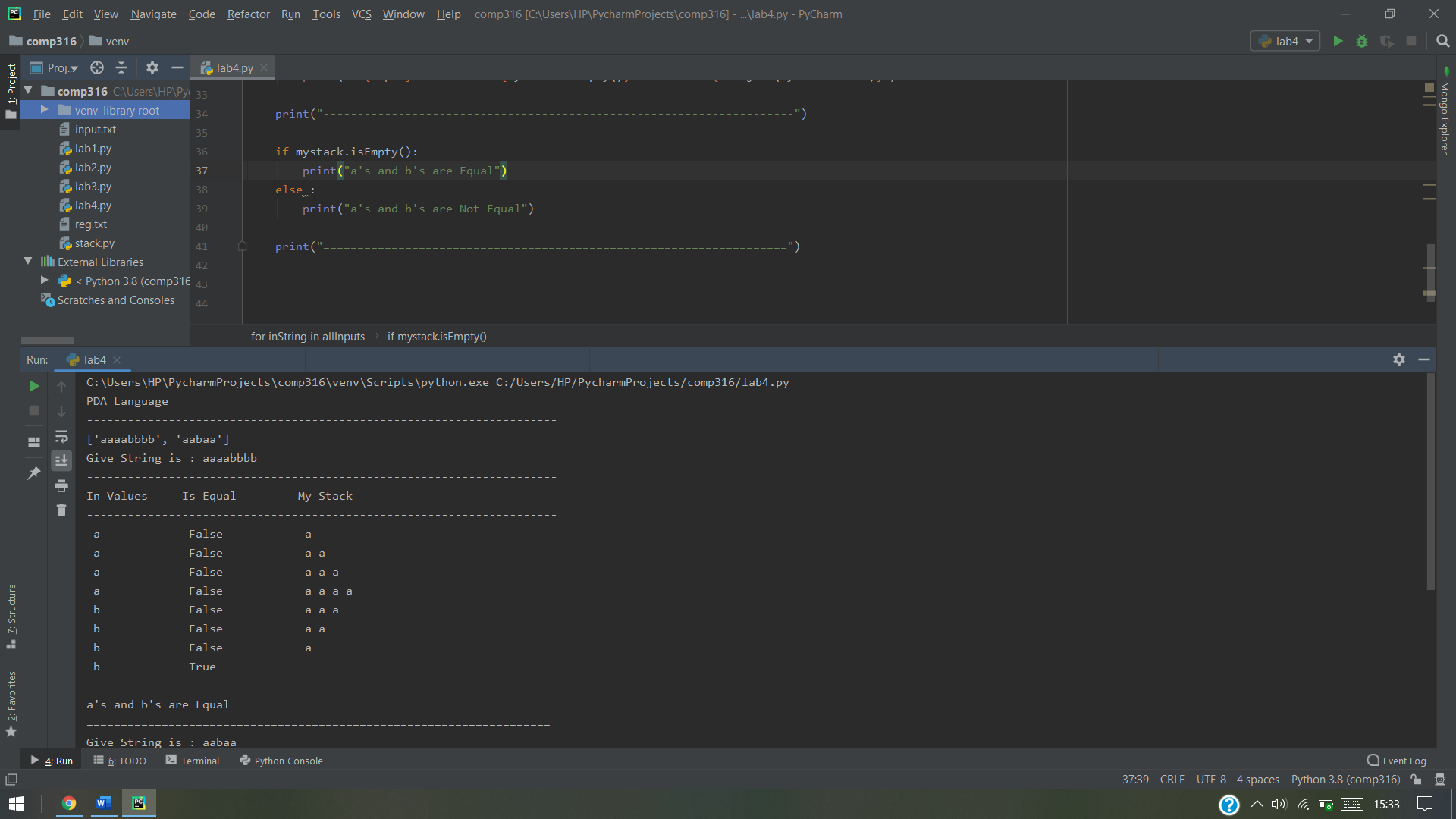
aaaabbbb

aabaa

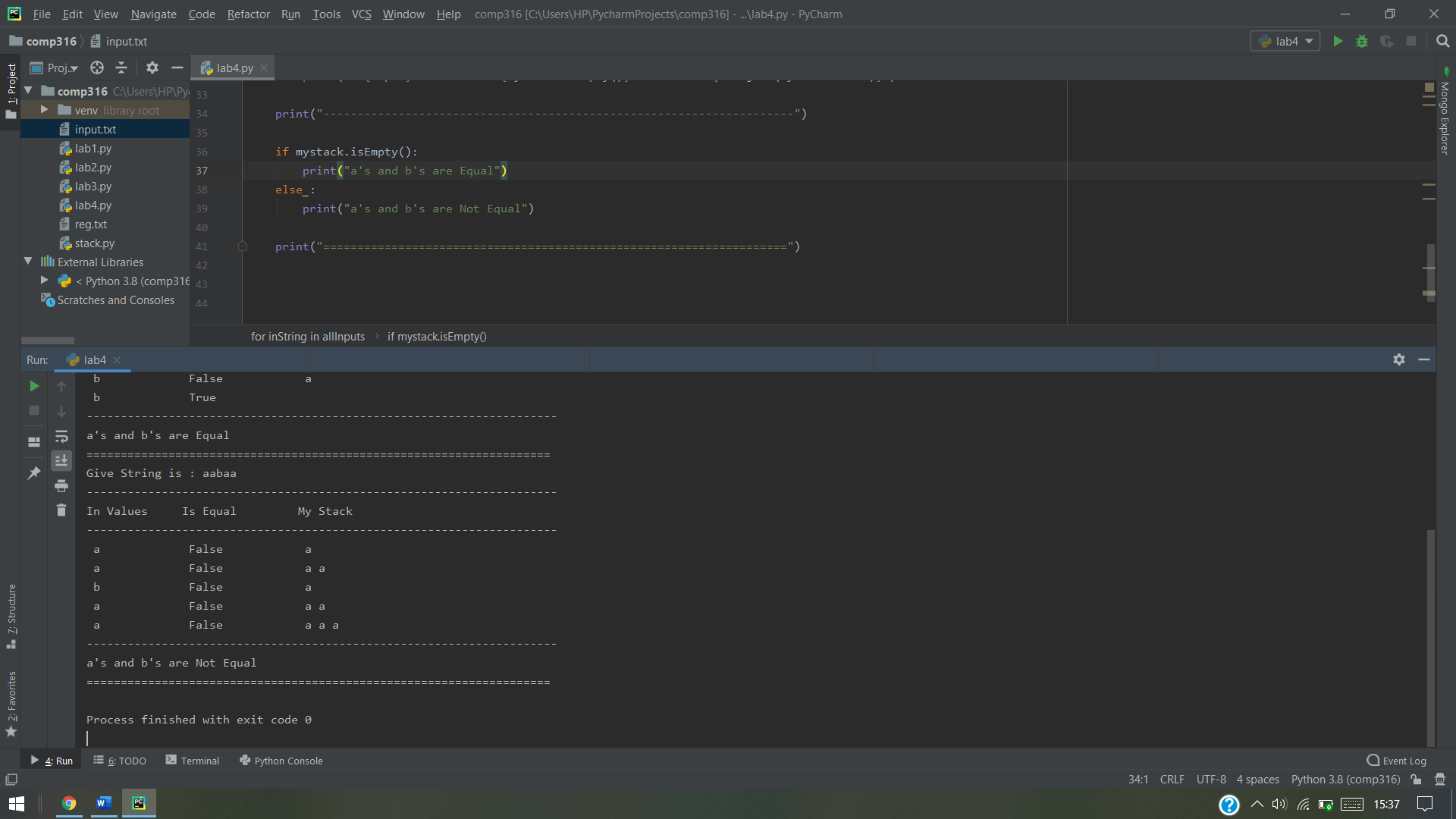
**Description:**

At first, we create text file where we keep the language for which we are going to implement PDA for. After that stack is created to store the data variables. The stack is implemented on a separate program. Then to implement the PDA, we call the stack from the file where we have created the stack and load it into our main program. Our main program scans the grammar and returns True only if the stack is empty. If the stack is empty than the PDA is implemented whereas if the stack still consists of variables the PDA is not implemented.

**Output:**



Output when PDA is implemented



Output when PDA is not implemented