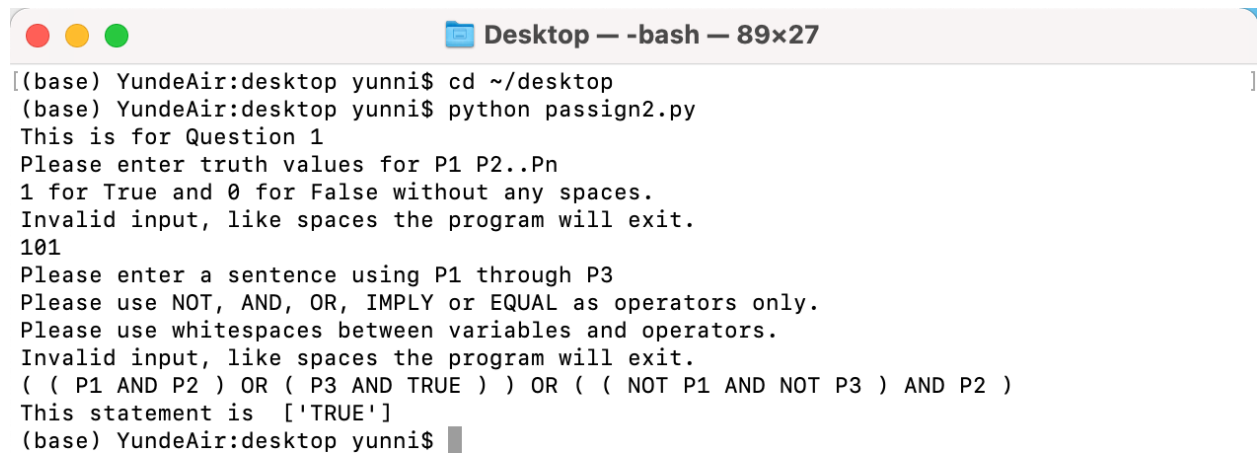Assignment 2

The format and how to run the code.

For Q1:

a) The program first takes a number of propositional variables (P1, P2, P3... Pn) as an input.

b) It then takes a statement involving these variables as an input.

c) The program parses the statement and gives "True" if the given assignment satisfies the sentence or "False" if it does not.

How to run the code:

a) First, users need to input the values for variables in Command Line.

b) Only 1 and 0 are valid input and there should be no whitespace between them.

c) Users need to input valid sentence. The Valid operator are NOT, AND, OR, IMPLY, EQUAL. All of the letters should in uppercase.

d) The program will exit, if the input has invalid operators.

```
● ● ●                    📖 Desktop — -bash — 89×27
[(base) YundeAir:desktop yunni$ cd ~/desktop                                        ]
(base) YundeAir:desktop yunni$ python passign2.py
This is for Question 1
Please enter truth values for P1 P2..Pn
1 for True and 0 for False without any spaces.
Invalid input, like spaces the program will exit.
101
Please enter a sentence using P1 through P3
Please use NOT, AND, OR, IMPLY or EQUAL as operators only.
Please use whitespaces between variables and operators.
Invalid input, like spaces the program will exit.
( ( P1 AND P2 ) OR ( P3 AND TRUE ) ) OR ( ( NOT P1 AND NOT P3 ) AND P2 )
This statement is  ['TRUE']
(base) YundeAir:desktop yunni$ ▌
```

Output for Q1

For Q2:

a) The program first takes a statement involving variables P1, P2, P3... as an input.

b) It then creates a truth table for this statement and outputs it.

c) The program will output the truth table whether the statement is a tautology, contradiction or contingency.

How to run the code:

a) First, users need to input propositional statement in the Command Line.

b) Variables and operators need a whitespace to separate them.

c) The program will exit, if the input has invalid operators.

```
● ● ●                    🖥 Desktop — -bash — 80×33
[(base) YundeAir:desktop yunni$ cd ~/desktop
[(base) YundeAir:desktop yunni$ python passign2.py
 This is for Q2
 Please enter a sentence using P1 through Pn.
 Please use NOT, AND, OR, IMPLY or EQAL as operators.
 Please use bracket and whitespaces between variables n operators.
 ( NOT P1 AND ( P1 OR P2 ) ) IMPLY P2
 P1  =  0
 P2  =  0
 --------------------------------------------
 The truth value for row  0  is  ['TRUE']
 --------------------------------------------
 P1  =  1
 P2  =  0
 --------------------------------------------
 The truth value for row  1  is  ['TRUE']
 --------------------------------------------
 P1  =  0
 P2  =  1
 --------------------------------------------
 The truth value for row  2  is  ['TRUE']
 --------------------------------------------
 P1  =  1
 P2  =  1
 --------------------------------------------
 The truth value for row  3  is  ['TRUE']
 --------------------------------------------
 This sentence is Tautology
 (base) YundeAir:desktop yunni$ ▊
```

Output for Q2-1

```
[(base) YundeAir:desktop yunni$ cd ~/desktop
[(base) YundeAir:desktop yunni$ python passign2.py
This is for Q2
Please enter a sentence using P1 through Pn.
Please use NOT, AND, OR, IMPLY or EQAL as operators.
Please use bracket and whitespaces between variables n operators.
P2 AND ( P1 IMPLY NOT P2 ) AND ( NOT P1 IMPLY NOT P2 )
P1  =  0
P2  =  0
-------------------------------------------
The truth value for row  0  is  ['FALSE']
-------------------------------------------
P1  =  1
P2  =  0
-------------------------------------------
The truth value for row  1  is  ['FALSE']
-------------------------------------------
P1  =  0
P2  =  1
-------------------------------------------
The truth value for row  2  is  ['FALSE']
-------------------------------------------
P1  =  1
P2  =  1
-------------------------------------------
The truth value for row  3  is  ['FALSE']
-------------------------------------------
This sentence is Contradiction
(base) YundeAir:desktop yunni$ ▊
```

Output for Q2-2

```
This is for Q2
Please enter a sentence using P1 through Pn.
Please use NOT, AND, OR, IMPLY or EQAL as operators.
Please use bracket and whitespaces between variables n operators.
        3. ( P1 IMPLY ( P2 IMPLY P3 ) ) IMPLY ( ( P1 IMPLY P2 ) IMPLY P3 )
P1  =  0
P2  =  0
P3  =  0
--------------------------------------------
The truth value for row  0  is  ['3.', 'FALSE']
--------------------------------------------
P1  =  1
P2  =  0
P3  =  0
--------------------------------------------
The truth value for row  1  is  ['3.', 'TRUE']
--------------------------------------------
P1  =  0
P2  =  1
P3  =  0
--------------------------------------------
The truth value for row  2  is  ['3.', 'FALSE']
--------------------------------------------
P1  =  1
P2  =  1
P3  =  0
--------------------------------------------
The truth value for row  3  is  ['3.', 'TRUE']
--------------------------------------------
P1  =  0
P2  =  0
P3  =  1
--------------------------------------------
The truth value for row  4  is  ['3.', 'TRUE']
--------------------------------------------
P1  =  1
P2  =  0
P3  =  1
--------------------------------------------
The truth value for row  5  is  ['3.', 'TRUE']
--------------------------------------------
P1  =  0
P2  =  1
P3  =  1
--------------------------------------------
The truth value for row  6  is  ['3.', 'TRUE']
--------------------------------------------
P1  =  1
P2  =  1
P3  =  1
--------------------------------------------
The truth value for row  7  is  ['3.', 'TRUE']
--------------------------------------------
This sentence is Contingency
(base) YundeAir:desktop yunni$
```

Output for Q2-3