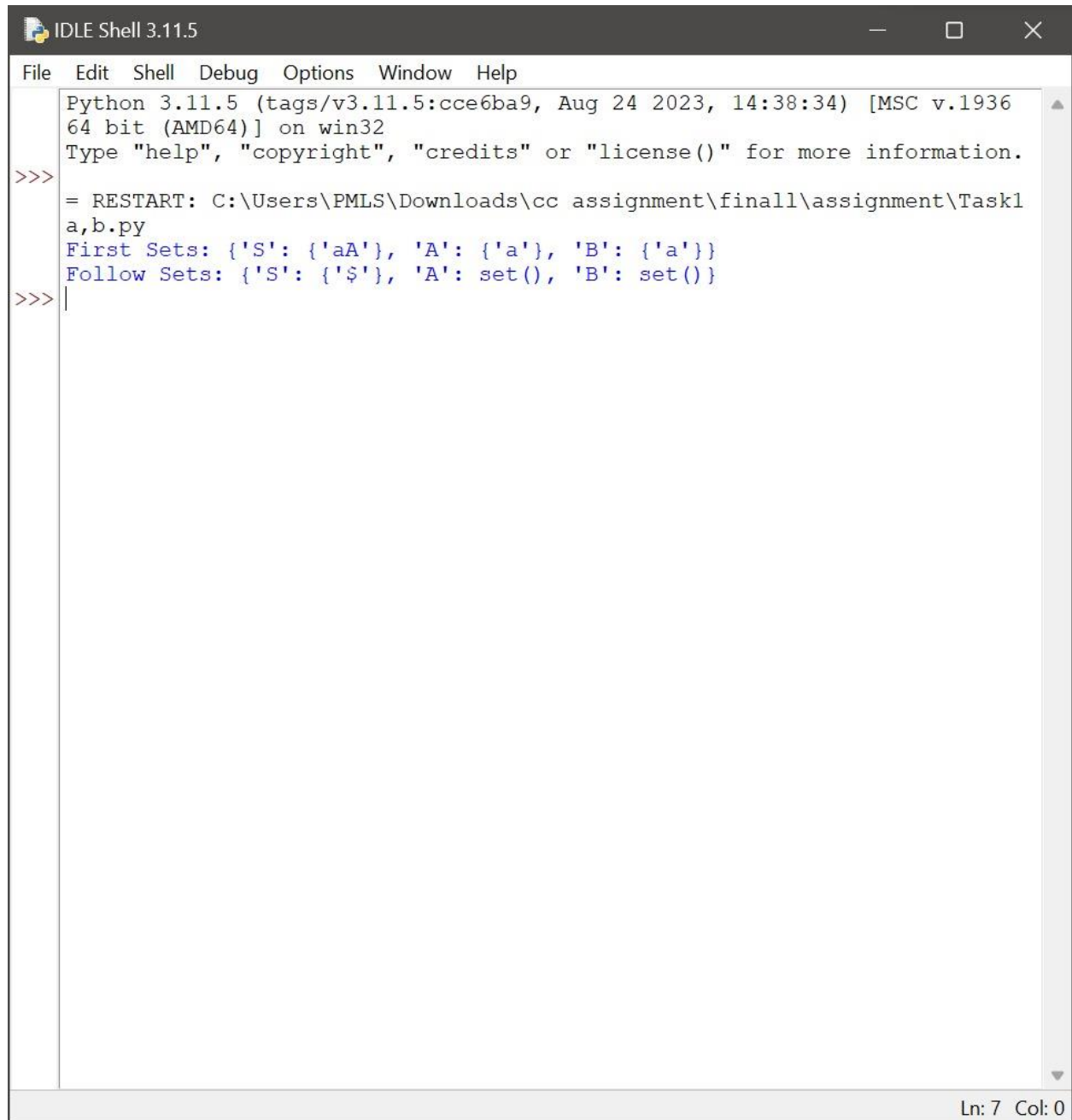


## Tasks Verification Screen Shots:

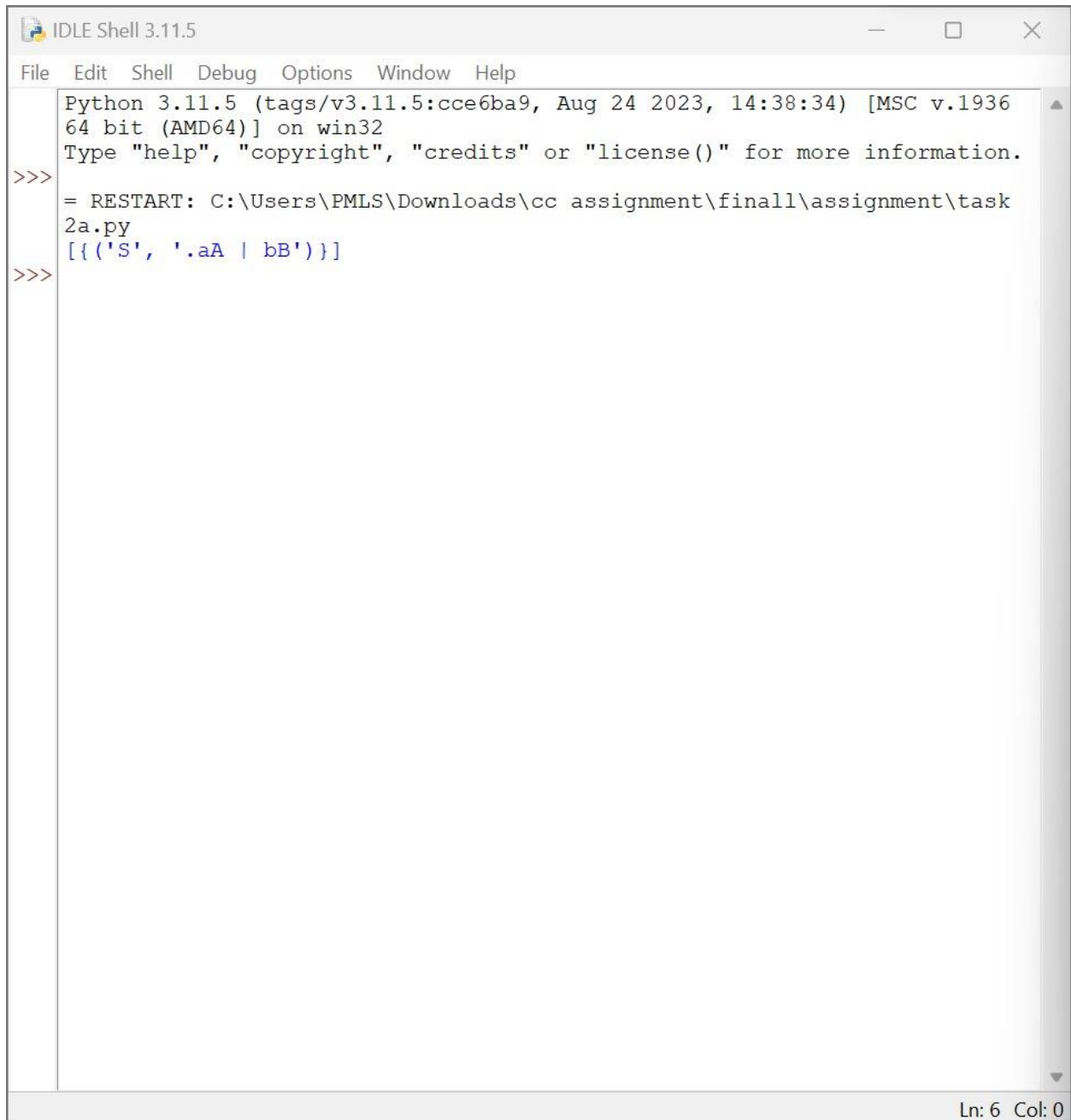
### Task\_1:



```
Python 3.11.5 (tags/v3.11.5:cce6ba9, Aug 24 2023, 14:38:34) [MSC v.1936
64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\PMLS\Downloads\cc assignment\finall\assignment\Task1
a,b.py
First Sets: {'S': {'aA'}, 'A': {'a'}, 'B': {'a'}}
Follow Sets: {'S': {'$'}, 'A': set(), 'B': set()}
>>> |
```

Ln: 7 Col: 0

## Task\_2a:

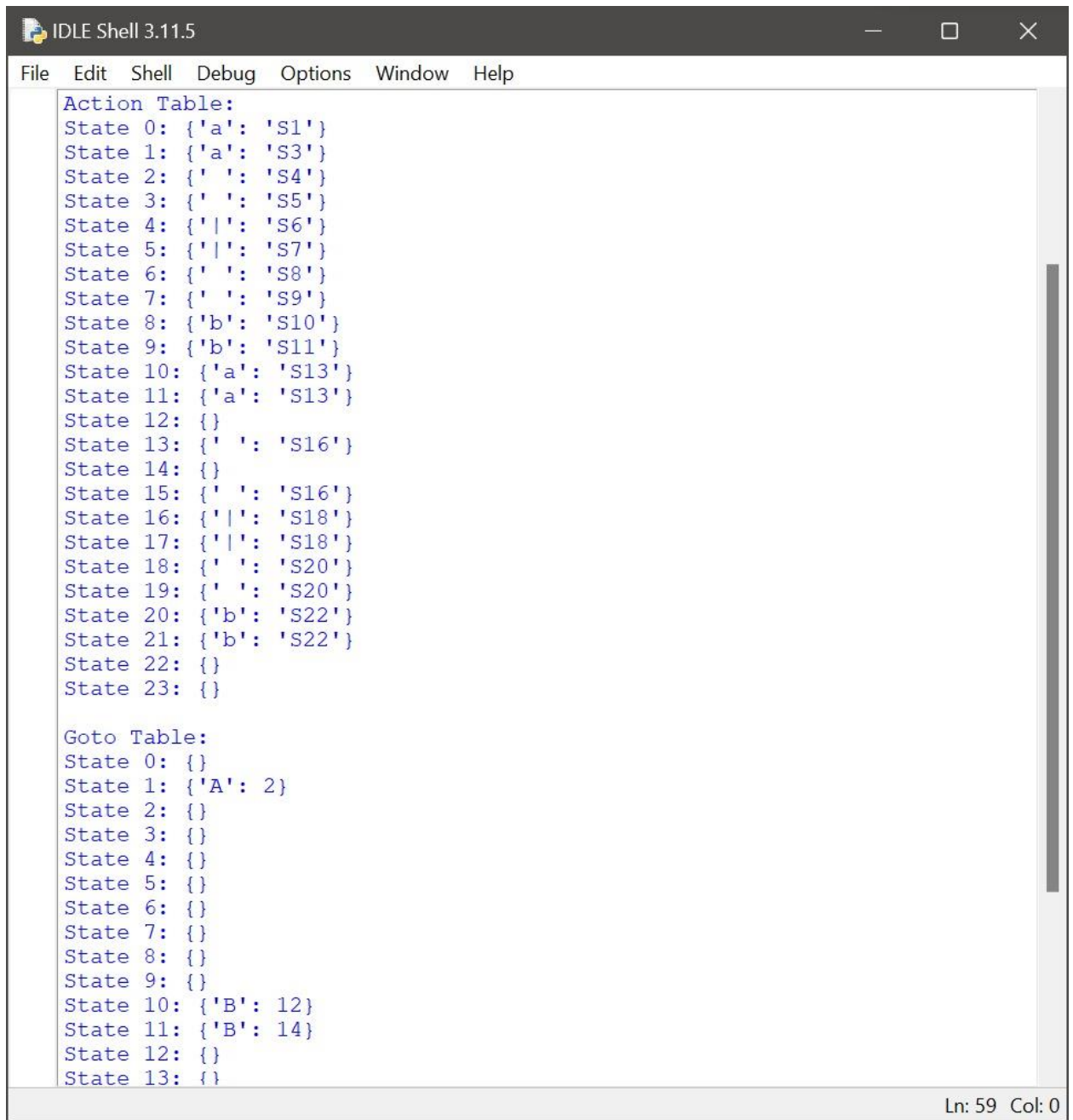


The screenshot shows the IDLE Shell 3.11.5 window. The title bar reads "IDLE Shell 3.11.5". The menu bar includes "File", "Edit", "Shell", "Debug", "Options", "Window", and "Help". The shell output displays the Python version and architecture: "Python 3.11.5 (tags/v3.11.5:cce6ba9, Aug 24 2023, 14:38:34) [MSC v.1936 64 bit (AMD64)] on win32". It also shows the prompt "Type 'help', 'copyright', 'credits' or 'license()' for more information." followed by a restart command: ">>> = RESTART: C:\Users\PMLS\Downloads\cc assignment\finall\assignment\task 2a.py". The code being executed is ">>> [({'S', '.aA | bB'})]". The status bar at the bottom right indicates "Ln: 6 Col: 0".

```
Python 3.11.5 (tags/v3.11.5:cce6ba9, Aug 24 2023, 14:38:34) [MSC v.1936
64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> = RESTART: C:\Users\PMLS\Downloads\cc assignment\finall\assignment\task
2a.py
>>> [({'S', '.aA | bB'})]
```

Ln: 6 Col: 0

## Task\_2b:



The screenshot shows a window titled "IDLE Shell 3.11.5" with a menu bar (File, Edit, Shell, Debug, Options, Window, Help) and a text area containing two tables. The first table, "Action Table", lists 24 states with their corresponding actions. The second table, "Goto Table", lists 14 states with their corresponding goto actions. The status bar at the bottom right indicates "Ln: 59 Col: 0".

```

IDLE Shell 3.11.5
File Edit Shell Debug Options Window Help

Action Table:
State 0: {'a': 'S1'}
State 1: {'a': 'S3'}
State 2: {' ': 'S4'}
State 3: {' ': 'S5'}
State 4: {'|': 'S6'}
State 5: {'|': 'S7'}
State 6: {' ': 'S8'}
State 7: {' ': 'S9'}
State 8: {'b': 'S10'}
State 9: {'b': 'S11'}
State 10: {'a': 'S13'}
State 11: {'a': 'S13'}
State 12: {}
State 13: {' ': 'S16'}
State 14: {}
State 15: {' ': 'S16'}
State 16: {'|': 'S18'}
State 17: {'|': 'S18'}
State 18: {' ': 'S20'}
State 19: {' ': 'S20'}
State 20: {'b': 'S22'}
State 21: {'b': 'S22'}
State 22: {}
State 23: {}

Goto Table:
State 0: {}
State 1: {'A': 2}
State 2: {}
State 3: {}
State 4: {}
State 5: {}
State 6: {}
State 7: {}
State 8: {}
State 9: {}
State 10: {'B': 12}
State 11: {'B': 14}
State 12: {}
State 13: {}

Ln: 59 Col: 0
```

### Task\_2c:

```

T -> T . F
F -> . F *
F -> . a
F -> . b

Result of GOTO computation:

GOTO ( I0 , E ) = I1
GOTO ( I0 , T ) = I2
GOTO ( I0 , F ) = I3
GOTO ( I0 , a ) = I4
GOTO ( I0 , b ) = I5
GOTO ( I1 , + ) = I6
GOTO ( I2 , F ) = I7
GOTO ( I2 , a ) = I4
GOTO ( I2 , b ) = I5
GOTO ( I3 , * ) = I8
GOTO ( I6 , T ) = I9
GOTO ( I6 , F ) = I3
GOTO ( I6 , a ) = I4
GOTO ( I6 , b ) = I5
GOTO ( I7 , * ) = I8
GOTO ( I9 , F ) = I7
GOTO ( I9 , a ) = I4
GOTO ( I9 , b ) = I5

LR(0) parsing table:

      a      +      *      b      $      E      T      F
I0      S4
I1  Accept  Accept  Accept  Accept  Accept  Accept  Accept  Accept
I2      R2      R2      R2      R2      R2      R2      R2      R2
I3      R4      R4      R4      R4      R4      R4      R4      R4
I4      R6      R6      R6      R6      R6      R6      R6      R6
I5      R7      R7      R7      R7      R7      R7      R7      R7
I6      S4      S5      S5      S5      S5      S5      S5      S5
I7      R3      R3      R3      R3      R3      R3      R3      R3
I8      R5      R5      R5      R5      R5      R5      R5      R5
I9      R1      R1      R1      R1      R1      R1      R1      R1

```

## Task3:

```
Python 3.11.5 (tags/v3.11.5:cce6ba9, Aug 24 2023, 14:38:34) [MSC v.1936  
64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.  
>>>  
= RESTART: C:\Users\PMLS\Downloads\cc assignment\finall\assignment\task  
3 bii.py  
Input the grammar productions  
Non terminals are represented by capital letters and terminals by small  
letters  
S -> abABC | DE  
A -> a |  $\epsilon$   
B -> bA |  $\epsilon$   
C -> c  
D -> d | (E)  
E -> e |  $\epsilon$   
  
-----First AND Follow Of Variables-----  
S  
    First:  {'a'}  
    Follow: {'$'}  
  
A  
    First:  {'a'}  
    Follow: {'|', 'b'}  
  
B  
    First:  {'b'}  
    Follow: {'c'}  
  
C  
    First:  {'c'}  
    Follow: {'|'}  
  
D  
    First:  {'d'}  
    Follow: {'e'}  
  
E  
    First:  {'e'}  
    Follow: {'$', ')}'  
  
Ln: 170 Col: 0
```

