

Μεταγλωτιστές 2020

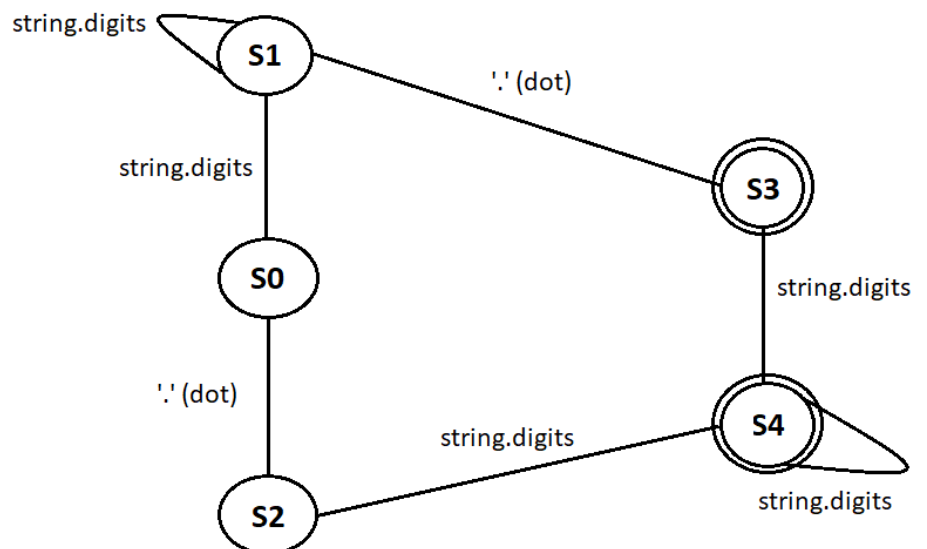
Προγραμματιστική Εργασία #1

Ονοματεπώνυμο: Λάζαρος Παπανικολάου

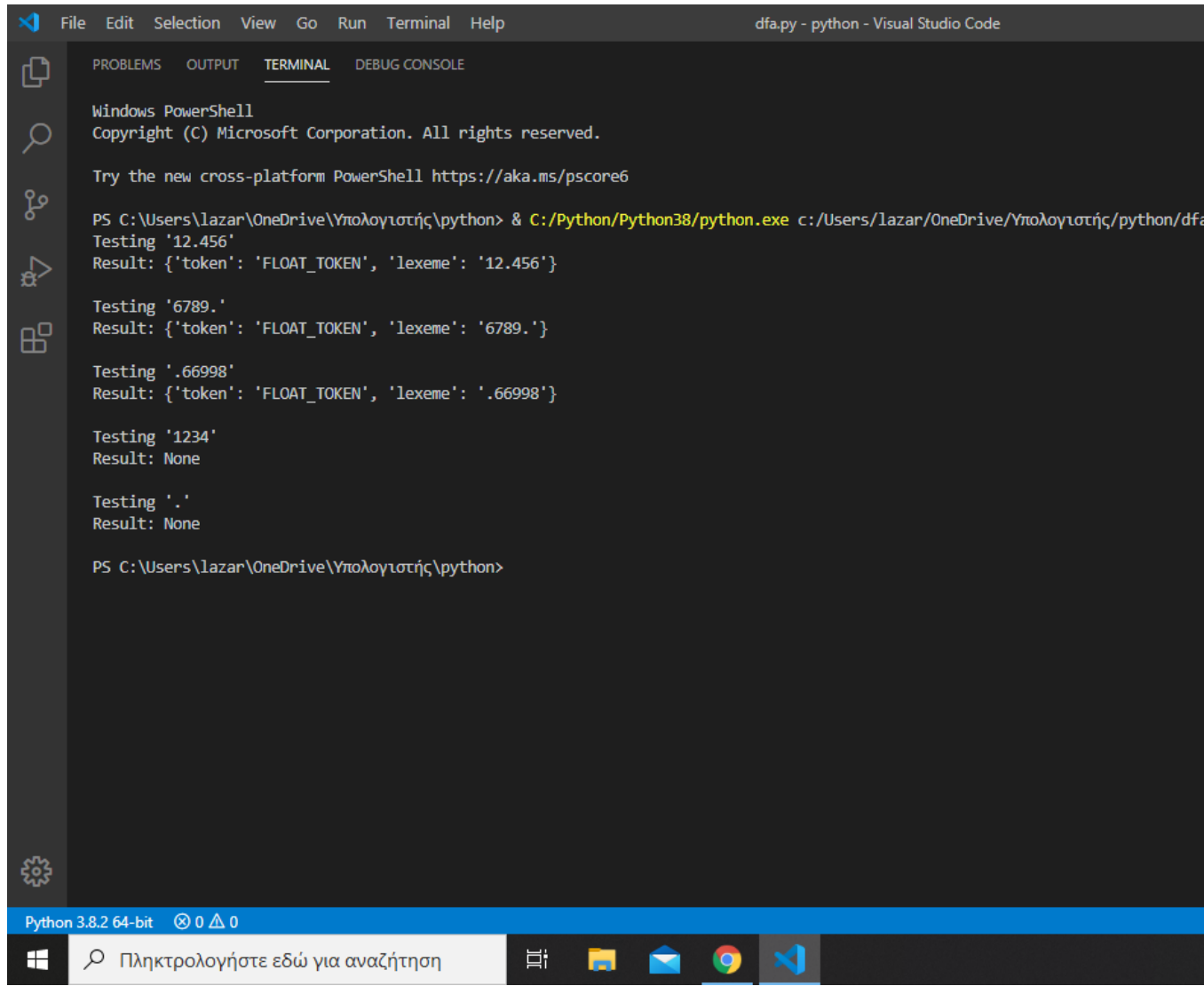
ΑΜ: Π2017085

Αναφορά της εργασίας μου σε pdf

Σχήμα του αυτομάτου πεπερασμένων
καταστάσεων (FA).



Αποτέλεσμα εξόδου (screenshot) του προγράμματος μου.



The screenshot shows a Windows PowerShell terminal window with a dark background. The title bar at the top reads "dfa.py - python - Visual Studio Code". The menu bar includes "File", "Edit", "Selection", "View", "Go", "Run", "Terminal", and "Help". The left sidebar contains icons for Explorer, Search, Source Control, Run and Debug, and Extensions. The main area displays the output of a Python script. The prompt is "PS C:\Users\lazar\OneDrive\Υπολογιστής\python>". The command executed is "& C:/Python/Python38/python.exe c:/Users/lazar/OneDrive/Υπολογιστής/python/dfa.py". The output shows several test cases for a lexer/parser. Each test case consists of a string being tested, followed by the result of the parsing. The results are JSON objects with 'token' and 'lexeme' fields. The test cases are: '12.456', '6789.', '.66998', '1234', and '.'. The results for '12.456', '6789.', and '.66998' are all 'FLOAT_TOKEN'. The results for '1234' and '.' are 'None'. The prompt returns to "PS C:\Users\lazar\OneDrive\Υπολογιστής\python>".

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\lazar\OneDrive\Υπολογιστής\python> & C:/Python/Python38/python.exe c:/Users/lazar/OneDrive/Υπολογιστής/python/dfa.py
Testing '12.456'
Result: {'token': 'FLOAT_TOKEN', 'lexeme': '12.456'}

Testing '6789.'
Result: {'token': 'FLOAT_TOKEN', 'lexeme': '6789.'}

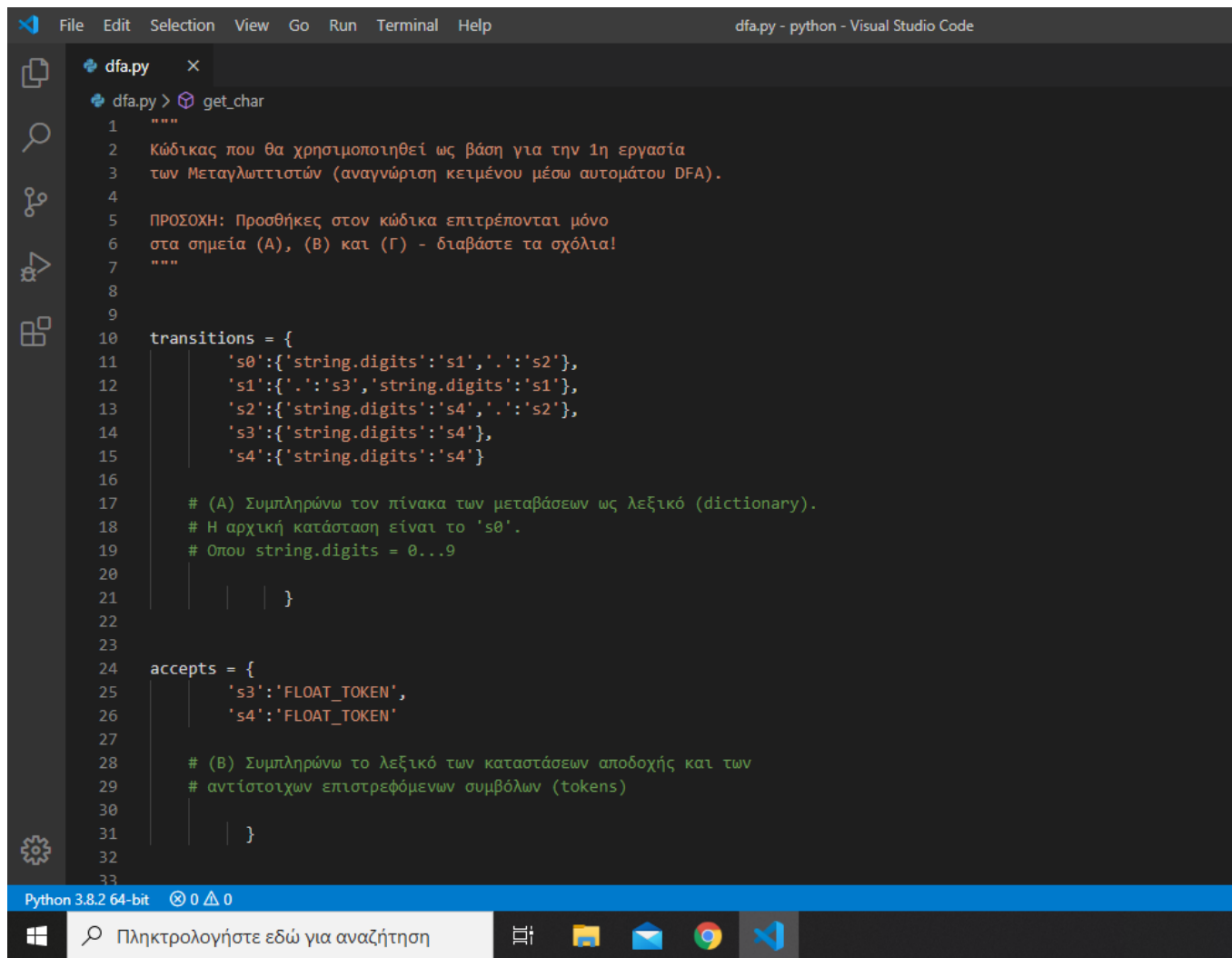
Testing '.66998'
Result: {'token': 'FLOAT_TOKEN', 'lexeme': '.66998'}

Testing '1234'
Result: None

Testing '.'
Result: None

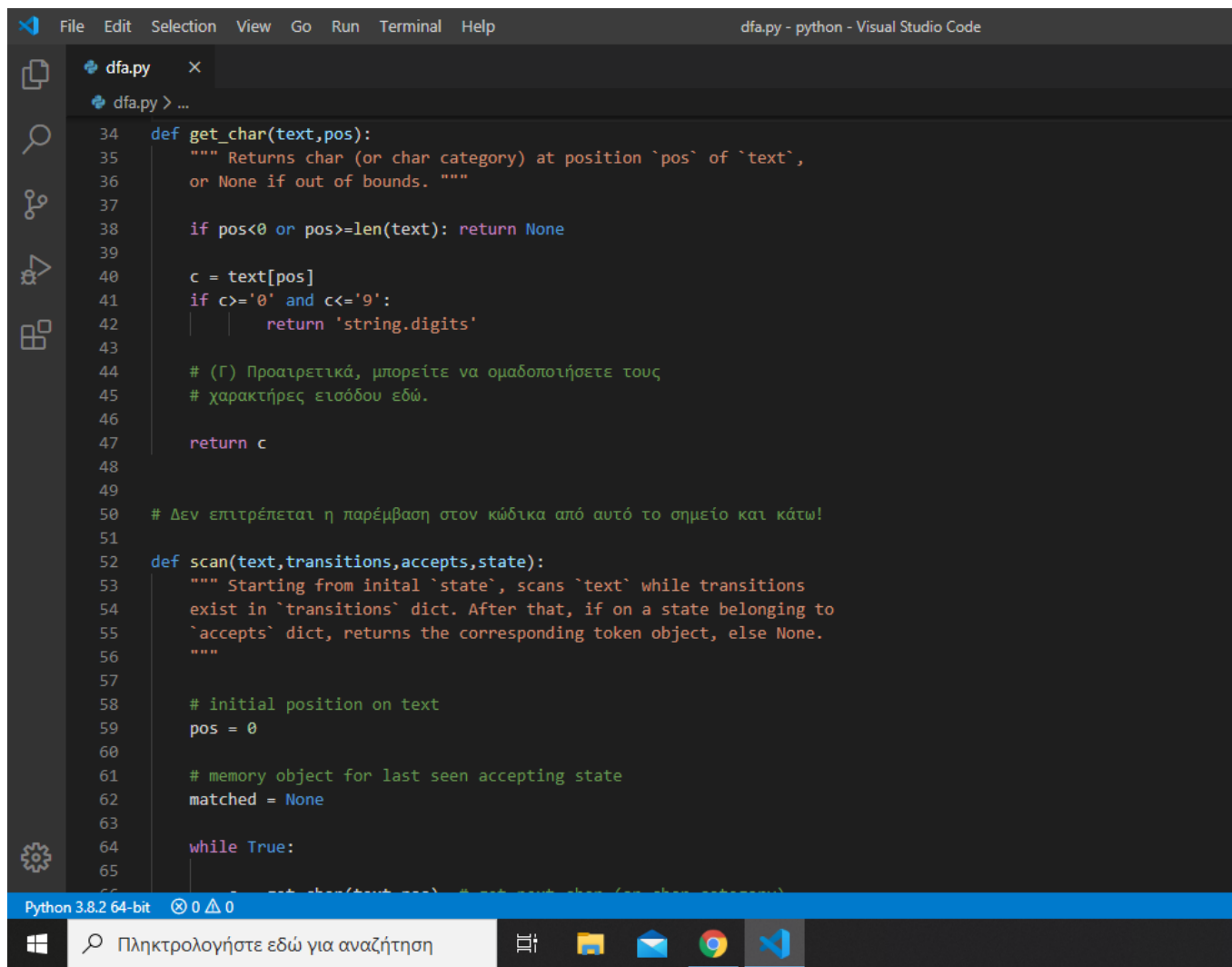
PS C:\Users\lazar\OneDrive\Υπολογιστής\python>
```

Κώδικας του προγράμματος μου.

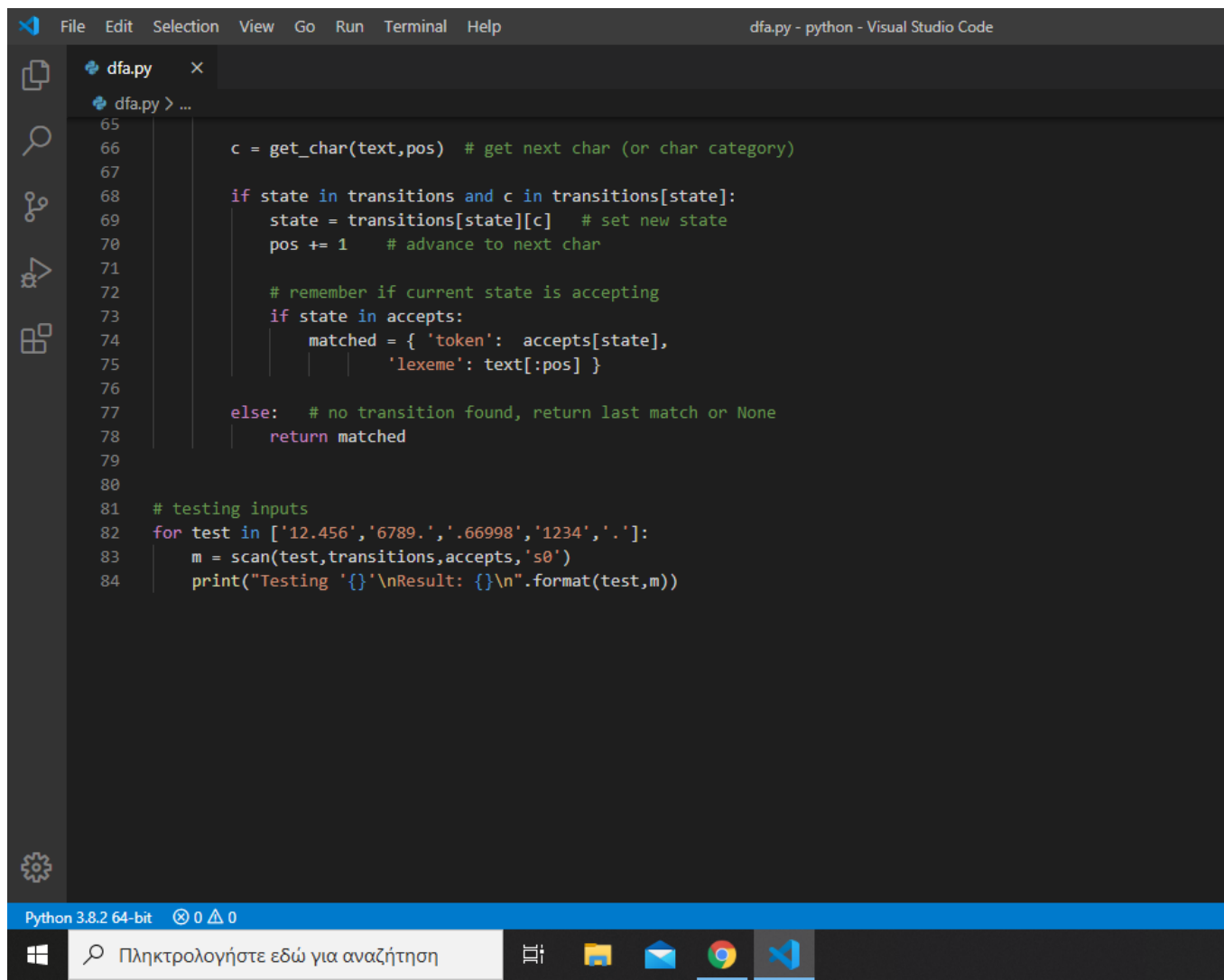


```
1 """
2 Κώδικας που θα χρησιμοποιηθεί ως βάση για την 1η εργασία
3 των Μεταγλωττιστών (αναγνώριση κειμένου μέσω αυτομάτου DFA).
4
5 ΠΡΟΣΟΧΗ: Προσθήκες στον κώδικα επιτρέπονται μόνο
6 στα σημεία (Α), (Β) και (Γ) - διαβάστε τα σχόλια!
7 """
8
9
10 transitions = {
11     's0': {'string.digits': 's1', '.': 's2'},
12     's1': {'.': 's3', 'string.digits': 's1'},
13     's2': {'string.digits': 's4', '.': 's2'},
14     's3': {'string.digits': 's4'},
15     's4': {'string.digits': 's4'}
16
17     # (Α) Συμπληρώνω τον πίνακα των μεταβάσεων ως λεξικό (dictionary).
18     # Η αρχική κατάσταση είναι το 's0'.
19     # Όπου string.digits = 0...9
20     |
21     |
22     }
23
24 accepts = {
25     's3': 'FLOAT_TOKEN',
26     's4': 'FLOAT_TOKEN'
27
28     # (Β) Συμπληρώνω το λεξικό των καταστάσεων αποδοχής και των
29     # αντίστοιχων επιστρεφόμενων συμβόλων (tokens)
30     |
31     |
32     }
33
```

Εικόνα 1.



Εικόνα 2.



```
65
66     c = get_char(text,pos) # get next char (or char category)
67
68     if state in transitions and c in transitions[state]:
69         state = transitions[state][c] # set new state
70         pos += 1 # advance to next char
71
72         # remember if current state is accepting
73         if state in accepts:
74             matched = { 'token': accepts[state],
75                         'lexeme': text[:pos] }
76
77     else: # no transition found, return last match or None
78         return matched
79
80
81 # testing inputs
82 for test in ['12.456','6789.','66998','1234','.']:
83     m = scan(test,transitions,accepts,'s0')
84     print("Testing '{}' \nResult: {}".format(test,m))
```

Εικόνα 3.

Για την υλοποίηση του προγράμματος βασίστηκα στις διαφάνειες του μαθήματος απο τα δοσμένα links καθώς και στην σελίδα GeeksforGeeks με το παρακατω link (<https://www.geeksforgeeks.org/python-string-digits/>).