

Θεωρία Υπολογισμών και Αυτομάτων

Εργασία #1 - Υλοποίηση αυτομάτου

Παυλίδης Παύλος
dai16035

28 Νοεμβρίου 2018

Η υλοποίηση του αυτόματου έγινε χρησιμοποιώντας την γλώσσα Python ενώ τα αρχεία για την εκτέλεση αλλά και ο κώδικας γράφτηκαν σε λειτουργικό Windows 10.

Πρόκειται για ένα πρόγραμμα που αποτελείται από τρεις συναρτήσεις:

- **readfile()** : Συνάρτηση που διαβάζει το αρχείο που έδωσε ο χρήστης με την περιγραφή του αυτόματου και αποθηκεύει τις πληροφορίες του στις αντίστοιχες μεταβλητές.

```
1 def readfile():
2     # This function reads the data from the input file and stores them
3     # inputs : -
4     # output : returns file's data
5
6     file = sys.argv[1]
7     with open(file, 'r') as f:
8         numberOfStatements = f.readline().rstrip('\n')
9         initialStatements = f.readline().rstrip('\n')
10        numberOfFinalStatements = f.readline().rstrip('\n')
11        finalStatements = [int(x) for x in f.readline().split()]
12        numberOfArrows = f.readline().rstrip('\n')
13
14        lines = []
15        for line in f: # read rest of lines
16            lines.append([x for x in line.split()])
17
18        nodeNames = set([])
19        for inner in lines:
20            nodeNames.add(inner[0])
21
22        for inner in nodeNames:
```

```

23         possibleMoves[inner] = []
24
25     for inner in lines:
26         possibleMoves[inner[0]].append([inner[1], inner[2]])
27
28     return numberOfStatements, initialStatements,
        numberOfFinalStatements, finalStatements, numberOfArrows,
        nodeNames

```

- numberOfStatements : Πλήθος καταστάσεων
- initialStatements : Αρχική κατάσταση του αυτόματου
- numberOfFinalStatements : Πλήθος τελικών καταστάσεων
- finalStatements : Η λίστα που είναι αποθηκευμένες οι τελικές καταστάσεις του αυτόματου
- numberOfArrows : Πλήθος μεταβάσεων (βελάκια)
- nodeNames : Λίστα που αποθηκεύει τις καταστάσεις του αυτόματου
- possibleMoves : Λίστα που αποθηκεύει τις μεταβάσεις που έχει κάθε κατάσταση του αυτόματου. Δηλαδή, κάθε κατάσταση πόσα βελάκια εξερχόμενα από αυτήν έχει

Η nodeNames μεταβλητή είναι ένα σύνολο όπου φιλοξενεί τις καταστάσεις. Είναι σύνολο προκειμένου κάθε τιμή να υπάρχει μόνο μια φορά. Έτσι μπορούμε να χρησιμοποιήσουμε τις τιμές της nodeNames ως keys στο dictionary possibleMoves. Αυτό θα έχει ως values τις μεταβάσεις που έχει το κάθε key. Έστω για παράδειγμα ότι η κατάσταση 2 έχει τις μεταβάσεις: a 3, a 2, b 5. Η possibleMoves θα έχει $\{2 : [['a', '3'], ['a', '2'], ['b', '5']]\}$.

- **main :** Η main φιλοξενεί την κλήση όλων των συναρτήσεων αλλά και την επικοινωνία του προγράμματος με τον χρήστη. Μόλις η ανάγνωση του αρχείου περιγραφής του αυτόματου τελειώσει, το πρόγραμμα ζητάει από τον χρήστη να πατήσει "1" αν θέλει να εισάγει αλφαριθμητικό, "2" αν θέλει να εισάγει χαρακτήρα - χαρακτήρα και "enter" αν θέλει να τερματίσει την εκτέλεση του. Σε αυτό σημείο πρέπει ωστόσο να τονίσουμε πως η επιλογή 2 είναι "εικονική". Ο χρήστης εισάγει όσους χαρακτήρες ή αλφαριθμητικά θέλει αλλά η εκτέλεση της συνάρτησης αξιολόγησης της εισόδου του θα εκτελεστεί αφού ο χρήστης τελειώσει με την είσοδο, δηλαδή αφού πατήσει 2 φορές στην σειρά "enter". Στην συνέχεια εκτελείται η συνάρτηση validateInput, το αποτέλεσμα της εκτυπώνεται στην κονσόλα και στην συνέχεια ο πρόγραμμα ξανά ρωτά τον χρήστη ποιον τρόπο εισαγωγής επιθυμεί να επιλέξει για την νέα είσοδο. Αν ο χρήστης διακόψει το πρόγραμμα, εμφανίζεται ανάλογο μήνυμα και το πρόγραμμα τερματίζεται.

```

1 if __name__ == "__main__":
2     possibleMoves = {}

```

```

3     nodeNames = set([])
4
5     numberOfStatements, initialStatements, numberOfFinalStatements,
        finalStatements, numberOfArrows, nodeNames = readfile()
6     inputletters = []
7
8     while True:
9         inputText = ""
10        chooseOption = input('Press 1 if you want to input a string or 2 if
            you want to input characters one at a '
11                                'time. If you want to exit please press enter: ')
12
13        if chooseOption == "":
14            break
15
16        if chooseOption == '1':
17            inputText = input('Enter a character or a string. If you want to
                exit please press enter: ')
18            if inputText == "":
19                break
20
21        elif chooseOption == '2':
22            print("Hit enter without any letter to finish character input: ")
23            while True:
24                tempInput = input('')
25                if tempInput == "":
26                    break
27                inputText = inputText + tempInput
28        else:
29            print("You gave invalid input. Please try again!!! ")
30            continue
31
32        if len(inputText) > 0:
33            inputLetters = [x for x in inputText]
34            if not (validateInput(inputLetters, initialStatements,
                finalStatements, possibleMoves)):
35                print("Invalid input \n")
36            else:
37                print("Valid input \n")
38
39        print("Program has been terminated")

```

- **validateInput()** : Συνάρτηση που αξιολογεί την είσοδο του χρήστη και επιστρέφει False αν η είσοδος του δεν οδηγεί σε τελική κατάσταση .

```

1 def validateInput(inputletters, initialStatements, finalStatements,
   possibleMoves):
2     # This function check if user's input will lead him to a final node or
   not
3     # inputs :
4     # inputletters : user's input
5     # initialStatements : the initial state of our automaton
6     # finalStatements : if user's input lead him to one of this nodes, then
   his input is valid
7     # possibleMoves : All possible moves in our automaton (all arrows)
8     # output : returns false if user's input is invalid
9
10    currentPosition = initialStatements
11
12    for letter in inputletters:
13
14        validLetter = False;
15        for moves in possibleMoves[currentPosition]:
16            # if there is a possible move from current position for this
   input letter
17            if letter == moves[0]:
18                currentPosition = moves[1]
19                validLetter = True
20
21        if not validLetter:
22            return False
23
24        # check if user's input lead him to a node that belongs to finalState
25        for finalState in finalStatements:
26            if finalState == int(currentPosition):
27                return True
28
29    return False

```

Αρχικά αποθηκεύουμε στην μεταβλητή `currentPosition` την τιμή της κατάστασης εκκίνησης. Η `currentPosition` θα κρατάει την θέση που οδηγηθήκαμε από μία μετάβαση. Για κάθε γράμμα της εισόδου κοιτάμε αν από την κατάσταση που βρισκόμαστε (αν δηλαδή από την `currentPosition`) υπάρχει μετάβαση. Αν ναι, η `validLetter` γίνεται `True` και ενημερώνεται η `currentPosition` με την τιμή της νέας κατάστασης. Αν ελέγξουμε το λεξικό μας και δεν βρούμε δυνατή μετάβαση, η `validLetter` θα έχει τιμή `False` και η `validateInput` θα επιστρέψει την τιμή `False`. Αν η `validLetter` δεν γίνει ποτέ `False`, τότε ελέγχουμε αν η `currentPosition` είναι κάποια από τις τελικές καταστάσεις και αν ναι, τότε επιστρέφουμε

True. Σημαντικό πλεονέκτημα της υλοποίησης αυτής είναι πως οτιδήποτε και αν δώσει ο χρήστης ως είσοδο, το πρόγραμμα δεν θα σπάσει αλλά θα εμφανίσει το μήνυμα λάθος εισόδου με το που φτάσει η στιγμή αξιολόγησης του λανθασμένου χαρακτήρα του χρήστη.

Τα αρχεία που δημιουργήθηκαν για τις δοκιμαστικές εκτελέσεις του προγράμματος βρέθηκαν μέσω αναζήτησης στο google σε μορφή εικόνας γι' αυτό και συμπεριλαμβάνονται στην παράδοση της εργασίας.

Στην συνέχεια ακολουθεί ένα screen shot που δείχνει την εκτέλεση του προγράμματος με όλες τις δυνατές επιλογές του χρήστη. Δηλαδή ο χρήστης να επιλέξει "1" ή "2" και η είσοδος του να είναι valid ή invalid:

```
C:\Users\Παυλος\PycharmProjects\automata_1>dai16035.exe input.txt
Press 1 if you want to input a string or 2 if you want to input characters one at a time. If you want to exit please press enter: 1
Enter a character or a string. If you want to exit please press enter: ab
Valid input

Press 1 if you want to input a string or 2 if you want to input characters one at a time. If you want to exit please press enter: 1
Enter a character or a string. If you want to exit please press enter: afg
Invalid input

Press 1 if you want to input a string or 2 if you want to input characters one at a time. If you want to exit please press enter: 2
Hit enter without any letter to finish character input:
b
c
e
Valid input

Press 1 if you want to input a string or 2 if you want to input characters one at a time. If you want to exit please press enter: 2
Hit enter without any letter to finish character input:
b
c
d
Invalid input

Press 1 if you want to input a string or 2 if you want to input characters one at a time. If you want to exit please press enter:
Program has been terminated

C:\Users\Παυλος\PycharmProjects\automata_1>
```

Σχήμα 1: Εκτέλεση του προγράμματος χρησιμοποιώντας το αρχείο input.txt