

# ΕΡΕΥΝΗΤΙΚΗ ΕΡΓΑΣΙΑ με τίτλο “Εισαγωγή στον Προγραμματισμό και την Ανάπτυξη Εφαρμογών με τη γλώσσα Python”

## ΚΡΕΜΑΛΑ – HANGMAN Οδηγίες

### Επιλογή άγνωστης λέξης

- 1ος τρόπος (Player vs Computer): Δημιουργία λίστας με λέξεις μέσα στο πρόγραμμα (περίπου 20 λέξεις). Αν θέλετε ελληνικές λέξεις, καλύτερα να χρησιμοποιήσετε κεφαλαία για να μην έχουμε πρόβλημα με τους τόνους, αλλιώς αγγλικές λέξεις. Το πρόγραμμα θα επιλέγει τυχαία μια λέξη από τη λίστα και αυτή τη λέξη θα πρέπει να μαντέψει ο παίκτης. Δείτε στο σκονάκι με τις λίστες το “Επιλογή ενός τυχαίου στοιχείου”
- 2ος τρόπος (Player vs Player): Χωρίς λίστα με λέξεις. Όταν ξεκινάει το πρόγραμμα, ο ένας παίκτης θα βάζει μια λέξη ( π.χ. word = input(‘Δώσε λέξη’) ) και ο άλλος παίκτης θα προσπαθεί να τη μαντέψει και στη συνέχεια θα γίνεται το αντίστροφο. Προσοχή: Επειδή με αυτόν τον τρόπο, η άγνωστη λέξη αποθηκεύεται όλη μαζί με μία μεταβλητή και όχι σε μία λίστα με ξεχωριστά τα γράμματά της όπως θέλουμε, θα πρέπει να μετατρέψετε αυτή τη μεταβλητή σε λίστα γράφοντας π.χ. word = list(word)

### Εμφάνιση άγνωστης λέξης με παύλες/αστεράκια:

Όποιον τρόπο και να επιλέξατε παραπάνω, έχετε τώρα μια λίστα με τα γράμματα της άγνωστης λέξης. Πρέπει λοιπόν να ‘εμφανίσετε’ την άγνωστη λέξη σαν μια σειρά από άγνωστους χαρακτήρες (παύλες, αστεράκια ή ότι άλλο θέλετε) για να τη μαντέψει ο παίκτης. Ο αριθμός των άγνωστων χαρακτήρων πρέπει φυσικά να είναι ίδιος με τον αριθμό των γραμμάτων της λέξης. Θα σας φανεί χρήσιμη η συνάρτηση len που μας δίνει το μέγεθος μιας λίστας ή μιας λέξης και ο τελεστής \* που μας δίνει μια λίστα που περιέχει πολλές φορές τα στοιχεία της αρχικής, δηλ. αν γράψουμε list = 5\*['a'] τότε το list θα είναι ίσο με ['a','a','a','a','a'].

### Game loop:

Μέχρι τώρα είχαμε την προετοιμασία για το κυρίως παιχνίδι. Τώρα είμαστε έτοιμοι να ξεκινήσουμε. Θα πρέπει λοιπόν να έχουμε μια επανάληψη μέσα στην οποία ο παίκτης θα βλέπει την άγνωστη λέξη σαν μια σειρά από άγνωστους χαρακτήρες, θα δίνει ένα γράμμα και το πρόγραμμα θα εμφανίζει τις θέσεις της λέξης οι οποίες περιέχουν το γράμμα ή θα εμφανίζει ένα μήνυμα αν δεν υπάρχει το γράμμα στη λέξη ή αν το γράμμα αυτό έχει ήδη δοθεί προηγουμένως. Χρειαζόμαστε μια άλλη λίστα η οποία θα κρατάει τα γράμματα που έχει δώσει ο παίκτης, ώστε το πρόγραμμα να ξέρει αν ο παίκτης έχει ξαναδώσει το ίδιο γράμμα στο παρελθόν. Επίσης χρειαζόμαστε μια μεταβλητή που θα μετράει τις αποτυχημένες προσπάθειες ώστε να ξέρουμε πότε θα χάσει ο παίκτης.

Άρα η λογική είναι κάπως έτσι (υπάρχουν φυσικά κι άλλοι τρόποι να το κάνετε):

- Όσο ο παίκτης δεν έχει βρει όλα τα γράμματα της λέξης και όσο δεν έχει χάσει:
- Ζήτησέ του να δώσει ένα γράμμα
- Αν το γράμμα δεν υπάρχει, εμφάνισε σχετικό μήνυμα, βάλε το γράμμα στη λίστα με τα γράμματα που έχουν ήδη δοθεί και πρόσθεσε μια αποτυχημένη προσπάθεια
- Αν το γράμμα έχει δοθεί ήδη, εμφάνισε σχετικό μήνυμα αλλά μην προσθέσεις αποτυχημένη προσπάθεια
- Αν το γράμμα υπάρχει στη λέξη, εμφάνισε σχετικό μήνυμα, βάλε το γράμμα στη λίστα με τα γράμματα που έχουν ήδη δοθεί και άλλαξε τη θέση (ή τις θέσεις) της άγνωστης λέξης στις οποίες υπάρχει το γράμμα ώστε να εμφανίζουν το γράμμα αυτό και όχι παύλες ή αστεράκια όπως πριν.
- Όταν τελειώσει η επανάληψη, έλεγξε αν ο παίκτης έχει βρει όλα τα γράμματα της λέξης και εμφάνισε σχετικό μήνυμα ή αν έχει χάσει και εμφάνισε σχετικό μήνυμα.
- Ρώτα αν θέλει να ξαναπαίξει και αν ναι, ξεκίνησε τη διαδικασία από την αρχή.

### **ΧΡΗΣΙΜΕΣ ΠΛΗΡΟΦΟΡΙΕΣ – ΣΥΝΔΕΣΜΟΙ**

- 1<sup>ο</sup> έγγραφο (Βασικές Εντολές Python): <http://pythonies.mysch.gr/csbasic.pdf>
- 2<sup>ο</sup> έγγραφο (Λίστες Python): <http://pythonies.mysch.gr/cslists.pdf>