

# ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΗ ΒΙΟΙΑΤΡΙΚΗ

## ΘΕΜΑΤΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Η/Υ - PROJECT 1

Βαρύτητα στον υπολογισμό του τελικού βαθμού : **30%**

Στα πλαίσια του project θα πρέπει να υλοποιήσετε ένα πρόγραμμα το οποίο θα σχεδιάζει την ημερήσια ξενάγηση του χρήστη σε μία πόλη. Ο χρήστης θα πρέπει να εισάγει διάφορα στοιχεία, όπως το όνομα της πόλης, το είδος των αξιοθέατων που τον ενδιαφέρουν κτλ. Το πρόγραμμα θα χρησιμοποιεί έτοιμα APIs, ώστε να ανακαλεί όλα τα απαιτούμενα στοιχεία. Είναι ήδη υλοποιημένες οι εξής τρεις συναρτήσεις : `get_city_landmarks`, `get_route_details` και `get_place_details`. Στο Παράρτημα αναλύεται ο τρόπος που λειτουργούν.

### 1. Εγκατάσταση προαπαιτούμενων (0.5)

Για να μπορέσετε να υλοποιήσετε το project στον υπολογιστή σας, θα πρέπει να εγκαταστήσετε την γλώσσα Python 3.6.1, τα πακέτα `googlemaps`, `python-google-places` και `wikipedia` και το περιβάλλον ανάπτυξης εφαρμογών Pycharm 2017.1. Πρέπει να ακολουθήσετε τα εξής βήματα :

α. Κάνετε λήψη του αρχείου εγκατάστασης **python-3.6.1.exe** από την ιστοσελίδα <https://www.python.org/downloads/> (κίτρινο button "Download Python 3.6.1") και το εκτελείτε, ώστε να εγκαταστήσετε την **Python 3.6.1** στον υπολογιστή σας. Στην αρχή της εγκατάστασης ενεργοποιήστε την επιλογή **Add Python 3.6 to PATH**

β. Ανοίγετε την **Γραμμή εντολών (Command line)** των Windows (<https://www.lifewire.com/how-to-open-command-prompt-2618089>) και εγκαθιστάτε τα απαραίτητα πακέτα της python πληκτρολογώντας την εξής εντολή: **pip install googlemaps python-google-places wikipedia**

γ. Κάνετε λήψη του αρχείου εγκατάστασης του **Pycharm Community Edition** από την ιστοσελίδα <https://www.jetbrains.com/pycharm/download/> και το εκτελείτε, ώστε να εγκαταστήσετε το περιβάλλον ανάπτυξης **Pycharm** στον υπολογιστή σας.

δ. Κάνετε αποσυμπίεση του αρχείου **design\_your\_tour.zip**. Μέσω του **Pycharm** χρησιμοποιείτε την επιλογή **Open** και επιλέγετε το directory **design\_your\_tour**. Εκτελείτε το αρχείο **main\_test.py** και αν όλα έχουν γίνει σωστά πρέπει να σας εμφανίσει το μήνυμα : **"Everything is OK !!!"**

### 2. Δημιουργία Google API key (0.5)

Για την χρήση των συναρτήσεων `get_city_landmarks()` και `get_route_details()` θα πρέπει να έχετε πρώτα συνδεθεί στην υπηρεσία **Google APIs**, να έχετε προμηθευθεί το **Google API key** σας και να έχετε ενεργοποιήσει τα απαραίτητα **APIs**. Η διαδικασία (υπάρχει και στο αρχείο **google\_apis.mp4**) είναι η εξής :

- Συνδέεστε στην ιστοσελίδα <https://console.developers.google.com>, χρησιμοποιώντας τον λογαριασμό σας στις υπηρεσίες της Google
- Επιλέγετε την ενότητα **Credentials**
- Επιλέγετε **Create** για να δημιουργήσετε νέο project
- Δίνετε ένα όνομα για το project σας, επιλέγετε **No** και **Yes** και πατάτε **Create**
- Επιλέγετε **Create Credentials - API Key**
- Αντιγράφετε το **string** που εμφανίζεται (είναι το προσωπικό σας API key)
- Πάτε στην ενότητα **Library** και στην κατηγορία **Google Maps API**, επιλέγετε το **Google Maps Distance Matrix API** και πατάτε το **Enable**
- Επαναλαμβάνετε το ίδιο για το **Google Places API Web Service** και το **Geocoding API**.

### 3. Δημιουργία συνάρτησης επιλογής είδους αξιοθέατων (0.5)

Να γράψετε μία συνάρτηση η οποία όταν εκτελείται, ρωτά τον χρήστη για το είδος των αξιοθέατων που τον ενδιαφέρουν και επιστρέφει την επιλεγμένη τιμή. Οι πιθανές τιμές είναι `aquarium`, `art_gallery`, `church`, `museum`, `park`, `zoo`. Εάν ο χρήστης δεν πληκτρολογήσει καμία από αυτές τότε του ζητείται να επιλέξει ξανά.

### 4. Ανάκληση αξιοθέατων για το Λονδίνο (0.5)

Να γράψετε ένα πρόγραμμα το οποίο να εμφανίζει τα ονόματα των σημαντικότερων μουσείων του Λονδίνου.

#### 5. Ανάκληση αξιοθέατων για μία πόλη που επιλέγει ο χρήστης (1)

Να γράψετε ένα πρόγραμμα το οποίο να ζητά από τον χρήστη το όνομα της πόλης που πρόκειται να επισκεφθεί καθώς και το είδος των αξιοθέατων που θα ήθελε να δει (χρησιμοποιώντας την συνάρτηση του ερωτήματος 3). Στην συνέχεια να εκτυπώνει μία λίστα με τα ονόματα των αξιοθέατων που ικανοποιούν τα παραπάνω κριτήρια.

#### 6. Ανάκληση αξιοθέατων και περιγραφών (1)

Να μεταβάλετε το πρόγραμμα του ερωτήματος 5 έτσι ώστε εκτός από τα ονόματα των αξιοθέατων να εκτυπώνει και μία σχετική περιγραφή για αυτά.

#### 7. Εμφάνιση περιγραφών μετά από επιλογή του χρήστη (1)

Να μεταβάλετε το πρόγραμμα του ερωτήματος 6 έτσι ώστε να εκτυπώνει μία αριθμημένη λίστα με τα ονόματα των αξιοθέατων και στην συνέχεια να ρωτά τον χρήστη για ποιο αξιοθέατο θέλει να δει την σχετική περιγραφή. Ο χρήστης θα δίνει τον αντίστοιχο αριθμό και το πρόγραμμα θα εκτυπώνει την περιγραφή. Στην συνέχεια η διαδικασία αυτή θα επαναλαμβάνεται μέχρι να πληκτρολογήσει ο χρήστης τον αριθμό 0.

#### 8. Δημιουργία λίστας αξιοθέατων προς επίσκεψη από τον χρήστη (1)

Να μεταβάλετε το πρόγραμμα του ερωτήματος 7 έτσι ώστε αφού εκτυπώνεται η περιγραφή για το επιλεγμένο αξιοθέατο να ρωτάει τον χρήστη αν θα ήθελε να το επισκεφθεί. Εάν η απάντηση είναι θετική τότε το πρόγραμμα θα προσθέτει το αξιοθέατο σε μία λίστα και μετά την έξοδο (με την πληκτρολόγηση του 0) θα εκτυπώνει την λίστα με τα αξιοθέατα προς επίσκεψη.

#### 9. Υπολογισμός απόστασης και χρόνου μετάβασης μεταξύ των αξιοθέατων (1)

Να μεταβάλετε το πρόγραμμα του ερωτήματος 8 έτσι ώστε να υπολογίζει την απόσταση και την ώρα που χρειάζεται να μεταβεί (περπατώντας) ο χρήστης από το ένα αξιοθέατο στο άλλο. Στην τελική λίστα των αξιοθέατων προς επίσκεψη να εκτυπώνει ανάμεσα στα αξιοθέατα και τις πληροφορίες αυτές.

#### 10. Χρήση ξενοδοχείου και υπολογισμός άφιξης σε κάθε αξιοθέατο (1.5)

Να μεταβάλετε το πρόγραμμα του ερωτήματος 9 έτσι ώστε να ρωτάει τον χρήστη σε ποιο ξενοδοχείο θα διαμένει και τι ώρα το πρωί θα ξεκινήσει την ξενάγηση στην πόλη. Στην τελική λίστα ξενάγησης θα περιλαμβάνεται και το ξενοδοχείο, ενώ θα τυπώνονται και οι ώρες που θα φτάνει ο χρήστης σε κάθε αξιοθέατο. (Θεωρείστε ότι η κάθε επίσκεψη θα διαρκεί 2 ώρες). Η ξενάγηση θα πρέπει να καταλήγει πάλι στο ξενοδοχείο. Αν ο χρήστης πρόκειται να φτάσει στο ξενοδοχείο μετά τις 22.00 τότε να τυπώνεται ένα σχετικό μήνυμα.

#### 11. Χρήση μέσων μαζικής μεταφοράς σε μεγάλες αποστάσεις (1.5)

Να μεταβάλετε το πρόγραμμα του ερωτήματος 10 έτσι ώστε το πρόγραμμα να ρωτάει επιπλέον τον χρήστη, ποια είναι η μέγιστη απόσταση που θέλει να περπατήσει μεταξύ δύο αξιοθέατων. Όταν η απόσταση μεταξύ δύο σημείων ξεπερνά αυτό το όριο τότε το πρόγραμμα θα πρέπει για αυτή την διαδρομή να επιλέγει την μετάβαση του χρήστη με την χρήση των μέσων μαζικής μεταφοράς. Στην εκτύπωση θα φαίνεται και ο τρόπος μετάβασης για κάθε διαδρομή.

#### 12. Αυτόματος σχεδιασμός ξενάγησης στην πόλη (bonus +1)

Να μεταβάλετε το πρόγραμμα του ερωτήματος 11 έτσι ώστε να καταρτίζει αυτόματα ένα πρόγραμμα επισκέψεων επιλέγοντας κάθε φορά το κοντινότερο επόμενο αξιοθέατο (ελέγχοντας όλους τους πιθανούς τρόπους μετάβασης). Προσθέτει τόσα αξιοθέατα έτσι ώστε ο χρήστης να γυρίσει στο ξενοδοχείο μέχρι και τις 22.00.

#### Παρατηρήσεις

- Θα πρέπει να υλοποιήσετε τα παραπάνω ζητούμενα και να υποβάλετε τα αντίστοιχα αρχεία (με ονόματα `main3.py`, `main4.py`, κτλ) συμπιεσμένα σε ένα **zip αρχείο** στην ενότητα **Εργασίες** στο **eclass**.
- Για να λειτουργούν οι έτοιμες συναρτήσεις θα πρέπει να αποσυμπίεσετε το αρχείο **design\_your\_tour.zip** και να τοποθετείτε τα αρχεία σας μέσα στο directory **design\_your\_tour**

- Επίσης θα πρέπει τα αρχεία σας να έχουν στην αρχή τους τις ίδιες δηλώσεις βιβλιοθηκών με το υπάρχον παράδειγμα `main_example.py`
- Για την καλύτερη κατανόηση των ζητούμενων προγραμμάτων, μπορείτε να παρακολουθήσετε την λειτουργία τους στο αρχείο `programs.mp4`

## ΠΑΡΑΡΤΗΜΑ

<code>get_city_landmarks(google_api_key, city, landmark_type)</code>	
Περιγραφή	Επιστρέφει μία λίστα με τις σημαντικότερες τοποθεσίες (τύπου <code>landmark_type</code> ) στην πόλη <code>city</code> . Για κάθε τοποθεσία επιστρέφει το όνομα της.
Ορίσματα	<code>string google_api_key</code> : Το ανακτάτε από το google developers console <code>string city</code> : Το όνομα της πόλης <code>string landmark_type</code> : Ο τύπος των αξιοθέατων που σας ενδιαφέρουν. Οι πιθανές τιμές είναι <code>aquarium, art_gallery, church, museum, park, zoo</code>
Αποτελέσματα	<code>[string name1, string name2,...]</code>
Σχόλια	Απαιτείται το προσωπικό σας <code>API key</code> από το <code>Google developers API</code> , και υπάρχει ένα ημερήσιο όριο χρήσης 1000 κλήσεων
Παράδειγμα	<code>get_city_landmarks("YOUR_API_KEY", "London", "museum")</code>  <code>['The British Museum', 'Natural History Museum', .....]</code>

<code>get_route_details(google_api_key, point1, point2, type_of_transport, when=None)</code>	
Περιγραφή	Επιστρέφει πληροφορίες σχετικά με την μετάβαση από το σημείο <code>point1</code> στο σημείο <code>point2</code> την χρονική στιγμή <code>when</code> μέσω του τρόπου <code>type_of_transport</code> . Επιστρέφει την σχετική απόσταση σε χιλιόμετρα, καθώς και την απαραίτητη χρονική διάρκεια σε λεπτά.
Ορίσματα	<code>string google_api_key</code> : Το ανακτάτε από το google developers console <code>string point1</code> : Το σημείο αφετηρίας <code>string point2</code> : Το σημείο τερματισμού <code>string type_of_transport</code> : Ο τρόπος μετακίνησης. Οι πιθανές τιμές είναι <code>driving, walking, bicycling, transit</code> <code>datetime when</code> : Η χρονική στιγμή της μετακίνησης
Αποτελέσματα	<code>[float distance, float duration]</code>
Σχόλια	Απαιτείται το προσωπικό σας <code>API key</code> από το <code>Google developers API</code> , και υπάρχει ένα ημερήσιο όριο χρήσης 2500 κλήσεων
Παράδειγμα	<code>get_city_landmarks("YOUR_API_KEY", 'The British Museum', 'Natural History Museum', 'transit')</code>  <code>[3.175, 16.15]</code>

<code>get_place_details(string place)</code>	
Περιγραφή	Επιστρέφει μία συνοπτική περιγραφή σχετικά με το αξιοθέατο <code>place</code> . Σε περίπτωση που αυτό είναι αδύνατο επιστρέφει το <code>string "-"</code> .
Ορίσματα	<code>string place</code> : Το όνομα του αξιοθέατου
Αποτελέσματα	<code>string description</code>
Σχόλια	-
Παράδειγμα	<code>get_place_details("Madame Tussauds London")</code>  <p>Madame Tussauds (UK /tuˈsɔːdz/, US /tuːˈsoʊz/; the family themselves pronounce it /ˈtuːsoʊ/) is a wax museum in London with smaller museums in a number of other major cities. It was founded by wax sculptor Marie Tussaud. It used to be known as "Madame Tussaud's"; the apostrophe is no longer used. Madame Tussauds is a major tourist attraction in London, displaying the waxworks of famous and historic peoples and also the film characters.</p>