

# ΣΚΑΡΤΣΙΛΑΣ ΔΑΜΙΑΝΟΣ

**1115201800180**

Παραδοτέα : **1<sup>η</sup> άσκηση** , έχουν γίνει τα ερωτήματα:

**a,b,c,d**(μόνο η κατα βαθος διασχιση και η εκτύπωση των κορυφών),**f,g**, ολόκληρα.

## Αρχεία υποβολής

GraphTypes.h

GraphInterface.c

GraphImplementation.c

Εντολή μεταγλώττισης και εκτέλεσης περιλαμβάνονται ως σχόλια στο GraphTypes.h

## GraphTypes.h

Περιλαμβάνει τις δομές αφηρημένου τύπου. Δηλαδή το **struct του γράφου** όπως ήταν στον κώδικα του github (απλά στο Edge προστέθηκε το startpoint), και επίσης το **struct της στοίβας Stack**, το οποίο χρησιμοποίησα στο **ερώτημα g** για να βρεθούν οι ισχυρά συνεκτικές συνιστώσες.

Περιλαμβάνονται και οι δηλώσεις των συναρτήσεων.

## GraphInterface.c

Σημειώσεις:

Οι συναρτήσεις Traverse και DepthFirst και PrintElement έχουν παρθεί απο τον δοσμένο κώδικα του github κάποιες απο τις οποίες έχω αλλάξει κάπως. Επίσης η συνάρτηση TraverseStacking είναι ή ίδια με την Traverse απλά στην TraverseStacking γίνεται Push στην στοίβα (ερώτημα

StrongComponents). Δηλαδή στη συνάρτηση StrongComponents εφαρμόζεται ο **αλγόριθμος του Kosaraju**, όπου γίνεται στην ουσία dfs διάσχιση στον αρχικό γράφο προσθέτοντας με τη βοήθεια της TraverseStacking και έπειτα dfs στον reversed γράφο με τη βοήθεια της Traverse.

στην **GraphReverse** χρησιμοποίησα την προσθήκη που έκανα (δηλαδή το startpoint) στο δοσμένο struct.

Έχουν υλοποιηθεί και οι συναρτήσεις StackInit, Pop και Push για το implementation της στοιβάς.

### **GraphImplementation.c**

Δημιουργία γράφου, εισαγωγή ακμών, εμφάνιση λίστας γειτνίασης, DFS, reversing, και τέλος εύρεση ισχυρά συνεκτικών συνιστωσών.