

Άσκηση 1 (προθεσμία 1/11/2018)

Γράψτε κώδικα-σκελετό που να δείχνει τα πρότυπα σχεδίασης (design patterns) Bridge και Visitor.

Ο κώδικας δεν χρειάζεται να παράγει αποτελέσματα (π.χ. οι μέθοδοι αρκεί απλά να τυπώνουν «κλήθηκε η μέθοδος ΧΨ»), αλλά πρέπει να έχει τέτοια δομή που τα πρότυπα σχεδίασης να έχουν νόημα. Για παράδειγμα για το πρότυπο Visitor μπορείτε να φανταστείτε ένα μεταγλωττιστή και να ονομάσετε τις κλάσεις σας αντίστοιχα (π.χ. AST, Stmt, Expr, PrintVisitor). **Χρησιμοποιήστε τη φαντασία σας για να βγάλετε σενάριο που ταιριάζει!**

Χρησιμοποιήστε C++ ή Java. Διαλέξτε τη γλώσσα που ξέρετε λιγότερο (δηλαδή αν ξέρετε καλά Java, διαλέξτε τη C++ και αντιστρόφως). Ο κώδικας πρέπει να μπορεί να περάσει από τον compiler και να τρέχει χωρίς δυναμικά σφάλματα.

Για να επιδείξετε την εφαρμογή, γράψτε μια main μέθοδο ή άλλες μεθόδους δοκιμασίας που να τρέχουν μόλις εκτελεστεί η main. Θα πρέπει να φαίνεται ποικιλομορφία στην εκτέλεση, που να δείχνει γιατί μπήκε το πρότυπο σχεδίασης. Π.χ. θα πρέπει η εκτέλεση να δείχνει δύο διαφορετικούς visitors, δύο διαφορετικά implementation classes για το bridge pattern, κτλ. πάνω στην ίδια δομή αντικειμένων. Δώστε αυτόματο τρόπο μεταγλώττισης και εκτέλεσης για διευκόλυνση του βαθμολογητή.

(Ξανά: δεν μας νοιάζει να υλοποιείται πραγματικά η λειτουργία που λέει το κάθε όνομα κλάσης ή μεθόδου, απλά α) το όνομα να βγάζει νόημα σαν κάτι που αντιστοιχεί σε πραγματική λειτουργία και β) η μέθοδος να τυπώνει πληροφοριακό μήνυμα όταν εκτελεστεί.)

Γράψτε μια σύντομη αναφορά που να απαντάει τουλάχιστον στις εξής ερωτήσεις:

- Γιατί ταίριαζε για τη χρήση αυτή το πρότυπο που διαλέξατε; Που υπήρχε μεταβλητότητα/ανάγκη ευελιξίας που επέβαλλε το πρότυπο;
- Δείξτε την ιεραρχία κλάσεων της εφαρμογής σας. Μερικές γραμμές κειμένου αρκούν, π.χ.:

```
class XY implements Visitable
class XYZ extends XY
class Z implements BImpl
class VisitorA implements XYVisitor
```

...

- Ποιες κλήσεις στον κώδικά σας είναι δυναμικές; Π.χ. στη δομή που φτιάξατε για επίδειξη της λειτουργικότητας, ξέρει ο γονιός-κόμβος τον δυναμικό τύπο του παιδιού-κόμβου όταν το visit του γονιού χρειάζεται visit του παιδιού;
- Ποια είναι τα πλεονεκτήματα/μειονεκτήματα της υλοποίησής σας; Π.χ. πόσο γενικό είναι το pattern για visit που υλοποιήσατε; (Πόσο γενικές είναι οι μέθοδοι σε σχέση με διαφορετικούς τύπους τιμών που μπορεί να επιστρέφει η visit, σε σχέση με exceptions, κτλ.;) Υπάρχει σημείο που να θέλει περισσότερη ευελιξία; Πώς θα τη δίνετε στη γλώσσα που διαλέξατε ή σε άλλη (π.χ. με templates/generics, με reflection, με multi-methods).

Η άσκηση θα βαθμολογηθεί συνοπτικά μόνο. Αν αναρωτηθείτε μόνοι σας για τα παραπάνω, θα μάθετε κάτι. Μην περιμένετε εκτενή σχολιασμό από το βοηθό.

Η άσκηση παραδίδεται με email, έως τα μεσάνυχτα (11:59μμ) της προθεσμίας.