

ΕΠΙΘΕΤΟ: ΟΝΟΜΑ:

ΕΞΑΜΗΝΟ: ΑΕΜ:

ΔΙΔΑΣΚΩΝ: Π. Κατσαρός

ΗΜ/ΝΙΑ ΠΑΡΑΔΟΣΗΣ ΑΣΚΗΣΗΣ ΕΡΓΑΣΤΗΡΙΟΥ: ΠΑΡΑΔΟΘΗΚΕ ΣΤΟΝ:

1. Θεωρείστε την κλάση:

```
class Date {
public:
    int _day;
    int _month;
    int _year;

    Date();
    Date(int d, int m, int y);    // δημιουργία ημερομηνίας από τις day, month, year
    void advance(long n);        // μετατόπιση της ημερομηνίας κατά n ημέρες
    int day();
    int month();
    int year();
    Date add_days(long n);        // υπολογίζει n ημέρες από την ημερομηνία
    long days_between(Date b);    // αριθμός ημερών μεταξύ της ημερομηνίας και της ημερομηνίας b
};
```

Στο πλαίσιο που ακολουθεί γράψτε πρόγραμμα που διαβάζει ένα σύνολο ημερομηνιών, τις ταξινομεί και τις εκτυπώνει. Κάντε προσπέλαση στα public δεδομένα - μέλη `_day`, `_month` και `_year` όποτε κρίνετε ότι αυτό χρειάζεται. (Μ 1.5)

Μετασχηματίστε τον ορισμό της κλάσης `Date` ώστε αυτός να εκφράζει αντικείμενα Ιουλιανών ημερομηνιών: μία Ιουλιανή ημερομηνία δίνεται από ένα `long`, που εκφράζει το συνολικό αριθμό ημερών από κάποια ημερομηνία αναφοράς (π.χ. γέννηση Χριστού). (M 1.0)

(α) Σχολιάστε την έκταση των αλλαγών που θα απαιτούσε η προσαρμογή του κώδικα ταξινόμησης που συγγράψατε, αν έπρεπε να χρησιμοποιηθεί η νέα αναπαράσταση της κλάσης `Date` και άρα να αντικατασταθούν οι αναφορές στα `_day`, `_month` και `_year` με αναφορές στο δεδομένο - μέλος των Ιουλιανών ημερομηνιών. (β) Τι θα έπρεπε να κάνετε αν εφαρμόζατε την αρχή της «απόκρυψης δεδομένων (data hiding)»; (γ) Πώς θα εξασφαλίζατε πρόσβαση στα δεδομένα - μέλη των αντικειμένων και πόσο πιο εύκολος θα ήταν ο μετασχηματισμός του κώδικα που συγγράψατε; (M 1.5)

2. Συσχετίστε κάθε χαρακτηριστικό από Α έως Ι της γλώσσας C++, με μία από τις φράσεις 1 - 10:
- | | |
|-----------------------------|-------------------------------------|
| Α. ΠΟΛΛΑΠΛΗ ΚΛΗΡΟΝΟΜΙΚΟΤΗΤΑ | ΣΤ. ΥΠΕΡΒΑΣΗ ΣΥΝΑΡΤΗΣΕΩΝ ΜΕΛΩΝ |
| Β. ΚΛΑΣΕΙΣ ΑΦΑΙΡΕΣΗΣ | Ζ. ΥΠΕΡΒΑΣΗ ΔΟΜΗΤΩΝ |
| Γ. ΦΙΛΕΣ ΣΥΝΑΡΤΗΣΕΙΣ | Η. ΜΕΘΟΔΟΙ ΕΓΚΑΤΑΣΤΑΣΗΣ Ή ΔΟΜΗΤΕΣ |
| Δ. <code>this</code> | Θ. ΠΑΡΑΜΕΤΡΟΠΟΙΗΣΙΜΑ ΤΜΗΜΑΤΑ ΚΩΔΙΚΑ |
| Ε. ΙΕΡΑΡΧΙΑ ΚΛΑΣΕΩΝ | Ι. ΔΙΑΧΕΙΡΙΣΗ ΕΞΑΙΡΕΣΕΩΝ |
1. Δημιουργία και αρχικοποίηση αντικειμένων.
 2. Δημιουργία αξιόπιστου προγράμματος.
 3. Πρόσβαση μέσα από συνάρτηση μέλος στη διεύθυνση αποθήκευσης του αντικειμένου.
 4. Δυνατότητα προσαρμογής σε διαφορετικές περιπτώσεις χρήσης της κλάσης/συνάρτησης.
 5. Δυνατότητα κλήσεων με τον ίδιο τρόπο συναρτήσεων-μελών για αντικείμενα μιας βασικής ή κάποιας παράγωγης κλάσης.
 6. Εναλλακτικές δυνατότητες δημιουργίας αντικειμένων.
 7. Επαναχρησιμοποίηση κώδικα που υλοποιείται σε ανεξάρτητες κλάσεις.
 8. Πρόσβαση στα ιδιωτικά και προστατευμένα δεδομένα περισσοτέρων της μιας ανεξάρτητων κλάσεων.
 9. Χρήση μόνο ως βάση κληρονομικότητας και όχι για τη δημιουργία νέων αντικειμένων.
 10. Προγραμματισμός παραλλαγών της ίδιας κλάσης που διαφοροποιούνται ως προς την υλοποίησή τους.

ΜΟΝΑΔΕΣ: 1.5

3. Τι πλεονεκτήματα έχει ο προγραμματισμός με γενίκευση (generic programming);

.....
.
.....
.....
.....
..

Δώστε ένα παράδειγμα χρήσης παραμετροποιήσιμου τμήματος κώδικα (template):