Αντικειμενοστραφής Προγραμματισμός 2017-2018

Εργασία 2

Σε αυτήν την εργασία θα εξοικειωθούμε με τις έννοιες της κλάσης, του αντικειμένου, των δημιουργών αντικειμένων (constructors), των καταστροφέων αντικειμένων (destructors), των συναρτήσεων, και των επιστρεφόμενων τιμών των συναρτήσεων. Είναι πολύ σημαντικό να αντιληφθούμε την δύναμη της αντικειμενοστρέφειας καθώς μας διευκολύνει στην περιγραφή των υπαρκτών και των ιδεατών αντικειμένων και την απλοποίηση των σχέσεων τους μέσα στον προγραμματισμό. Μην ξεχνάτε ότι οτιδήποτε εφαρμόζετε στην C++ με παρόμοιο τρόπο εφαρμόζεται και σε άλλες αντικειμενοστραφείς γλώσσες όπως είναι η C#, η Java ή η Python. Επίσης, θα διασκεδάσουμε βάζοντας Σαμουράι να μονομαχήσουν.

Ερωτήματα:

1. Ορίστε μια κλάση Samurai που αναπαριστά έναν πολεμιστή Σαμουράι. Κάθε αντικείμενο χρειάζεται τις εξής μεταβλητές:
    - **name**:τύπος string --> θα πρέπει να κάνετε include τις βιβλιοθήκες <iostream> και <string>
    - **numberOfWins:** τύπος int
    - **numberOfInjuries:** τύπος int
    - **numberOfDuels:** τύπος int
2. Ορίστε τον constructor της Samurai κλάσης ώστε να δουλεύει η δήλωση:

    Samurai a(“Giorgos”, 0, 0, 0), b(“Giannis”, 1, 2, 3);

    Όταν καλείται ο constructor θα πρέπει να εκτυπώνεται στην οθόνη “**Samurai ready for duel.**”
3. Για κάθε μεταβλητή της Samurai θα πρέπει να ορίσετε μια public συνάρτηση που να ξεκινάει με get στον ορισμό της. Συγκεκριμμένα, η getName() να επιστρέφει το όνομα του Samurai. Αντίστοιχα θέλουμε και τις getNumberOfWins(), getNumberOfInjuries() και getNumberOfDuels(). **ΠΡΟΣΟΧΗ** στον τύπο επιστροφής της κάθε συνάρτησης. Αυτές οι συναρτήσεις ονομάζονται στην αργκό των προγραμματιστών getters.
4. Για κάθε μεταβλητή της Samurai θα πρέπει να ορίσετε μια public συνάρτηση που να ξεκινάει με set στον ορισμό της και να δέχεται ένα όρισμα ίδιου τύπου με τη μεταβλητή της κλάσης. π.χ. η public void setName(string a){

name = a;

} Αυτές οι συναρτήσεις ονομάζονται στην αργκό των προγραμματιστών setters και χρησιμοποιούνται για να θέτουν στα πεδία της κλάσης καινούριες τιμές.

5. Ορίστε μια public συνάρτηση **printSamuraiDescription(Samurai x)** τύπου **void,** δηλαδή η συνάρτηση πέρνει ως όρισμα ένα αντικείμενο τύπου Samurai**.** Η συνάρτηση θα πρέπει να τυπώνει το εξής μήνυμα: **Samurai name:W Duels:X Wins:Y Injuries:Z** όπου **W,X,Y,Z** το όνομα του Samurai, ο αριθμός των μονομαχιών, ο αριθμός των νικών και ο αριθμός των τραυμάτων αντίστοιχα.
6. Ορίστε τον destructor της Samurai κλάσης. Όταν καλείται ο destructor θα πρέπει να εκτυπώνεται στην οθόνη **"Samurai X deleted.".** Όπου X το όνομα του Samurai.

   Όλες οι παραπάνω συναρτήσεις αποτελούν συναρτήσεις της κλάσης Samurai. Αυτό σημαίνει ότι όταν δημιουργήσουμε ένα αντικείμενο τύπου Samurai έστω a τότε θα πρέπει να μπορούμε να καλούμε τις συναρτήσεις μέσα από αυτό το αντικείμενο με τη δήλωση a.*όνομα_συνάρτησης* π.χ. a.printSamuraiDescription() για να εκτυπώσουμε τα στοιχεία του Samurai a.
7. Έξω από την κλάση Samurai ορίστε μια συνάρτηση duel τύπου **void** ως εξής: **void duel(Samurai &a, Samurai &b, bool winner)**

- Με το που θα καλείται η συνάρτηση duel θα πρέπει να αυξάνεται το numberOfDuels των συμμετέχοντων Samurai κατά 1.
- Στη συνέχεια:
  - Αν το όρισμα winner είναι true τότε ο Samurai a κερδίζει και το numberOfWins του θα πρέπει να αυξηθεί κατά 1 ενώ ο Samurai b χάνει και το numberOfInjuries του θα πρέπει να αυξηθεί κατά 1.
  - Αν το όρισμα number είναι false κερδίζει ο Samurai b και χάνει ο Samurai a με αντίστοιχη αύξηση των μεταβλητών τους.
  - Τέλος: θα πρέπει να υπάρχει έλεγχος έτσι ώστε αν κάποιος νικητής με αυτήν την μονομαχία οι νικές του ξεπερνάνε τις 10 θα πρέπει να εκτυπώνεται: "Samurai X is duelist.", όπου X το όνομα του Samurai.

Διευκρινίσεις/Παρατηρήσεις

• Όλα τα κενά αποτελούνται από ένα space και στα μηνύματα που χρειάζεται να εκτυπωθούν μην βάζετε endline χαρακτήρα.

• Το παραδοτέο να ονομαστεί XXXX.cpp και στην πρώτη γραμμή του να περιέχει το //XXXX όπου XXXX ο Α.Μ. σας.

• Το XXXX.cpp να έχει απαραίτητα σε σχόλια τη συνάρτηση main(). Για τη βαθμολόγηση δεν χρειαζόμαστε τη main, αλλά για τις μελλοντικές διευκρινίσεις πάνω στον κώδικά σας θα ήταν θεμιτό να υπάρχει μέσα στο παραδοτέο σε σχόλια προκειμένου να υπάρχει καλύτερη κατανόηση του τρόπου που εργαστήκατε.

• Οι τρεις παραπάνω διευκρινίσεις είναι πολύ σημαντικές γιατί οι εργασίες θα βαθμολογηθούν εν μέρει αυτόματα.

• Μη παραδώσετε εργασία που δεν κάνει compile σε τουλάχιστον έναν από τους compilers: Dev C++ (σε Windows συστήματα), g++(σε linux/MacOS συστήματα). Αν έχετε την δυνατότητα, προτιμήστε τον g ++. Εργασίες που δεν μεταγλωττίζονται βαθμολογούνται αυτόματα με μηδέν.

• Η εργασία είναι ατομική. Σε περίπτωση αντιγραφής, όλες οι ίδιες/παρόμοιες εργασίες θα

μηδενιστούν και οι παραβάτες θα θεωρηθούν μετεξεταστέοι στο Εργαστήριο, δλδ δεν θα μπορέσουν να πάρουν μέρος στην γραπτή εξέταση για αυτή την ακαδημαϊκή χρονιά.

Καλή επιτυχία