

4^η Προγραμματιστική Εργασία

Ημερομηνία παράδοσης: 13.11.2022

Συζήτηση λύσης - επίλυση αποριών: 16.11.2022

Στόχοι: Εξοικείωση με τη δήλωση και υλοποίηση συναρτήσεων, χρήση κλάσεων ως παραμέτρους και ως τύπους επιστροφής, τοπικές μεταβλητές.

Άσκηση 1.

Να γραφεί μια συνάρτηση `hmsToSec` η οποία παίρνει τρεις τιμές τύπου `int` ως ορίσματα —οι οποίες αντιστοιχούν σε ώρες, λεπτά και δευτερόλεπτα, αντίστοιχα— και επιστρέφει τον ισοδύναμο χρόνο σε δευτερόλεπτα (ως τύπο `long`). Στην υλοποίησή σας, η δήλωση και η υλοποίηση της συνάρτησης να είναι στο ίδιο αρχείο.

Στο αρχείο αυτό, να υλοποιηθεί η συνάρτηση-οδηγός `main()` η οποία ζητά από το χρήστη τρεις τιμές χρόνου (ώρες, λεπτά και δευτερόλεπτα), καλεί τη συνάρτηση και εμφανίζει την τιμή των δευτερολέπτων που επιστρέφει.

Άσκηση 2.

Στη διάλεξη είδατε πώς μπορεί κάποιος να γράψει συναρτήσεις οι οποίες έχουν ως παραμέτρους αντικείμενα. Το παράδειγμα που χρησιμοποιήθηκε ήταν αυτό της συνάρτησης `add()`, η οποία παίρνει σαν παραμέτρους (`const`, με αναφορά) δύο αντικείμενα της κλάσης `Fraction` και επιστρέφει ένα νέο αντικείμενο που αντιστοιχεί στο άθροισμά τους.

Στην εργασία αυτή, θα πρέπει να επεκτείνετε περαιτέρω τον κώδικα αυτό. Κατεβάστε από το E-Course το αρχείο `Fraction.cpp`, το οποίο θα βρείτε στο συνοδευτικό υλικό του δεύτερου μέρους των διαλέξεων και επεκτείνετε την υλοποίησή με τις παρακάτω συναρτήσεις:

`Fraction subtract(const Fraction &f, const Fraction &g)`

Επιστρέφει ένα νέο αντικείμενο της κλάσης `Fraction` το οποίο αντιστοιχεί στη διαφορά $f-g$.

`Fraction multiply(const Fraction &f, const Fraction &g)`

Επιστρέφει ένα νέο αντικείμενο της κλάσης `Fraction` το οποίο αντιστοιχεί στο γινόμενο $f*g$.

`Fraction divide(const Fraction &f, const Fraction &g)`

Επιστρέφει ένα νέο αντικείμενο της κλάσης `Fraction` το οποίο αντιστοιχεί στο πηλίκιο f/g .

`bool equal(const Fraction &f, const Fraction &g)`

Επιστρέφει `true` αν τα f και g έχουν την ίδια δεκαδική τιμή.

`bool isGreater(const Fraction &f, const Fraction &g)`

Επιστρέφει `true` αν η δεκαδική τιμή του f είναι (αυστηρώς) μεγαλύτερη αυτής του g .

`bool isSmaller(const Fraction &f, const Fraction &g)`

Επιστρέφει `true` αν η δεκαδική τιμή του f είναι (αυστηρώς) μικρότερη αυτής του g .

Επεκτείνετε την υλοποίηση της συνάρτησης `main()` έτσι ώστε να ελέγξετε την ορθότητα όλων των παραπάνω συναρτήσεων.

Στις γραμμές 63-69 του αρχείου που κατεβάσατε από το E-Course περιγράφεται ένας τρόπος υπερφόρτωσης του τελεστή `+` (κάνοντας μια απλή μετονομασία της συνάρτησης `add`), ο οποίος σας επιτρέπει να μπορείτε να προσθέσετε δύο αντικείμενα της κλάσης `Fraction` με τον ίδιο ακριβώς τρόπο με τον οποίο θα προσθέτατε δύο βασικούς τύπους δεδομένων, π.χ., `int`. Ένας ισοδύναμος τρόπος είναι ο ακόλουθος:

```
Fraction operator+(const Fraction &f, const Fraction &g) {  
    return add(f, g);  
}
```

Εισάγετε τον παραπάνω κώδικα στην υλοποίηση σας και προσπαθήστε να καταλάβετε τι κάνει. Επίσης, επεκτείνετε την υλοποίηση της συνάρτησης `main()` έτσι ώστε να υπολογίσετε το άθροισμα των αντικειμένων `f` και `g` χρησιμοποιώντας την ακόλουθη γραμμή κώδικα (όπως περιγράφεται στις γραμμές 63-69 του αρχείου που κατεβάσατε από το E-Course):

```
Fraction p = f + g;
```

Αφού κατανοήσετε τον τρόπο λειτουργίας της συνάρτησης `operator+` η οποία σας επιτρέπει να προσθέσετε δύο αντικείμενα τύπου `Fraction` χρησιμοποιώντας τον τελεστή `+`, να υλοποιήσετε με αντίστοιχο τρόπο τις παρακάτω συναρτήσεις:

```
Fraction operator-(const Fraction &f, const Fraction &g)  
Επιστρέφει ένα νέο αντικείμενο της κλάσης Fraction το οποίο αντιστοιχεί στη διαφορά f-g.  
Fraction operator*(const Fraction &f, const Fraction &g)  
Επιστρέφει ένα νέο αντικείμενο της κλάσης Fraction το οποίο αντιστοιχεί στο γινόμενο f*g.  
Fraction operator/(const Fraction &f, const Fraction &g)  
Επιστρέφει ένα νέο αντικείμενο της κλάσης Fraction το οποίο αντιστοιχεί στο πηλίκο f/g.  
bool operator==(const Fraction &f, const Fraction &g)  
Επιστρέφει true αν τα f και g έχουν την ίδια δεκαδική τιμή.  
bool operator>(const Fraction &f, const Fraction &g)  
Επιστρέφει true αν η δεκαδική τιμή του f είναι (αυστηρώς) μεγαλύτερη αυτής του g.  
bool operator<(const Fraction &f, const Fraction &g)  
Επιστρέφει true αν η δεκαδική τιμή του f είναι (αυστηρώς) μικρότερη αυτής του g.
```

Επεκτείνετε περαιτέρω την υλοποίηση της συνάρτησης `main()` έτσι ώστε να ελέγξετε την ορθότητα όλων των παραπάνω τελεστών. Τέλος, εμπλουτίστε τον κώδικα σας με σχόλια πριν την παράδοση του.