

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
Κ10:ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΑΦΗΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

Εξετάσεις 8 Σεπτεμβρίου 2020

1. (α') Αναφέρετε δύο βασικές λειτουργικότητες που πρέπει να παρέχει μια γλώσσα προγραμματισμού στον προγραμματιστή προκειμένου να μπορεί να θεωρηθεί γλώσσα αντικειμενοστραφούς προγραμματισμού.
- (β') Αιτιολογώντας την απάντησή σας —με όποιον τρόπο σας εξυπηρετεί—, δώστε το αποτέλεσμα της εκτέλεσης του παρακάτω προγράμματος C++.

```
#include<iostream>
using namespace std;
class A{ int data;
public:
    A(): data(10) { cout << "A new A has been constructed" << endl; }
    virtual ~A() { cout << " An A will be destructed with data: " << data << endl; }
    virtual void f1() { cout << "A::f1() is executing " << endl; data++; }
    void f2() { cout << "A::f2() is executing " << endl; f1(); } };

class B: public A{ int data;
public:
    B(): data(100) { cout << "A new B has been constructed" << endl; }
    ~B() { cout << " A B will be destructed with data: " << data << endl; }
    void f1() { cout << "B::f1() is executing " << endl; data++; }
    virtual void f2() { cout << "B::f2() is executing " << endl; f1(); } };

class C: public B{ int data;
public:
    C(): data(1000) { cout << "A new C has been constructed" << endl; }
    ~C() { cout << " A C will be destructed with data: " << data << endl; }
    void f1() { cout << "C::f1() is executing " << endl; data++; }
    void f2() { cout << "C::f2() is executing " << endl; f1(); } };

int main(){

    A* pa = new A(); pa->f1(); pa->f2(); delete pa;

    pa = new B(); pa->f1(); pa->f2(); delete pa;

    pa = new C(); pa->f1(); pa->f2(); delete pa; }
```

2. Θεωρήστε τον παρακάτω κόσμο ζωτικών που ευλογούν και, σε κάποιες περιπτώσεις, καταριούνται το ένα το άλλο. Τα ζωτικά διακρίνονται σε καλά ζωτικά και κακά ζωτικά.

Κάθε ζωτικό (creature)

- έχει ένα βαθμό ευεξίας (`happiness_degree`)
- μπορεί να ευλογήσει (`spell`) ένα άλλο. Σε αυτή την περίπτωση, ο βαθμός ευεξίας του ζωτικού που ευλογείται αυξάνεται κατά μια μονάδα.
- υφίσταται κατάρα (`gets_curse`). Σε αυτή την περίπτωση, ο βαθμός ευεξίας του μειώνεται κατά μία μονάδα. Αν ο βαθμός ευεξίας ενός ζωτικού μηδενιστεί, να τυπώνεται το μήνυμα "What a miserable world"

Αν ο βαθμός ευεξίας ενός ζωτικού γίνει ίσος με μια σταθερά μέγιστου βαθμού, κοινή για όλα τα ζωτικά (`MAX_HAPPINESS`), δεν μπορεί να τροποποιηθεί —ούτε να μειωθεί ούτε να αυξηθεί—.

Κάθε καλό ζωτικό (`good_creature`)

- έχει επιπλέον ένα μετρητή βαθμού καλοσύνης (`kindness_degree`)
- όταν ευλογεί (`spell`), επιπλέον της συμπεριφοράς του ως ζωτικό, αυξάνεται κατά μία μονάδα και ο βαθμός καλοσύνης του. Για να συμβούν αυτά, πρέπει ο βαθμός ευεξίας του να είναι πάνω από μηδέν.

Κάθε κακό ζωτικό (`bad_creature`)

- έχει επιπλέον ένα μετρητή βαθμού κακίας (`badness_degree`)
- όταν ευλογεί (`spell`), επιπλέον της συμπεριφοράς του ως ζωτικό, μειώνεται κατά μία μονάδα και ο βαθμός κακίας του.
- μπορεί να καταραστεί (`curse`) ένα άλλο ζωτικό, οτιδήποτε και να είναι αυτό και, τότε, αυξάνεται κατά μία μονάδα ο βαθμός κακίας του. Το ζωτικό στο οποίο απευθύνεται η κατάρα, υφίσταται κατάρα (`gets_curse`). Για να συμβούν αυτά, πρέπει ο βαθμός κακίας του κακού ζωτικού να είναι πάνω από μηδέν και ο βαθμός ευεξίας του μικρότερος από τη σταθερά μέγιστου βαθμού ευεξίας.

Υλοποιήστε σε C++ τις κλάσεις που αναπαριστούν τον παραπάνω κόσμο. Οι αρχικές τιμές δεδομένων των αντικειμένων των κλάσεων να παίρνουν τιμές που να δίδονται κατά τη δημιουργία των αντικειμένων. Σημείωση: Μπορείτε να κάνετε δικές σας παραδοχές για τον κόσμο αυτό αρκεί να μην έρχονται σε αντίφαση με την παραπάνω περιγραφή.