

S09 – Functii virtuale si clase abstracte

Alin Zamfiroiu

alin.zamfiroiu@csie.ase.ro

Funcții virtuale

- Funcțiile virtuale reprezintă de fapt un pointer la funcție.
- Fiind un pointer la funcție apelul efectiv se face la momentul rularii și nu la momentul compilării.
- Apare astfel termenul de late-binding. Ceea ce înseamnă că la momentul compilării nu se știe încă ce metodă va fi apelată, ci doar la momentul execuției se știe sigur metoda.

Funcții virtuale

- O funcție virtuală este introdusă prin cuvântul cheie: `virtual`.
- În clasa `Student` să se declare o funcție virtuală `bugetStudent()` care returnează o valoare `float`.
- În clasa `Student` funcția returnează valoarea `0`;

Functii virtuale

- In clasa StudentBursier si clasa StudentTaxa sa se supraincarce metoda bugetStudent().
- In clasa StudentBursier, metoda returneaza bursa acestuia.
- In clasa StudentTaxa, metoda returneaza taxa pe care acesta o plateste, cu semnul minus(-), deoarece pentru student taxareprezinta un cost.

Functii virtuale

- In programul apelator (main) se declara un pointer la Student.
- Acestui pointer i se de pe rand adresa unui Student, StudentBursier, StudentTaxa. Pentru fiecare caz se apeleaza metoda bugetStudent().
- Modificati in clasa Student metoda bugetStudent (sa nu mai fie virtuala ci o metoda normala). Ce diferente apar?

Funcții virtuale pure

- O funcție virtuală pură reprezintă o funcție virtuală fără corp.
- O funcție virtuală pură reprezintă un pointer inițializat cu `nullptr`.
- O funcție virtuală pură este inițializată cu 0 și nu i se implementează corp.

Functii virtuale pure

- In clasa Student, sa se modifice metoda `bugetStudent()`, astfel incat sa fie o functie virtuala pura.
- Ce diferente apar la apelare?
- De ce nu putem sa initializam un obiect de tipul Student?

Clase abstracte

- Clasa abstracta este clasa care are cel puțin o funcție virtuală pură.
- Clasa abstractă nu poate fi instantiată.
 - **De ce?**

Clase abstracte

- O clasa abstracta poate contine attribute, metode normale si constructori.
- **De ce** sunt utile attributele intr-o clasa abstracta daca nu se pot initializa obiecte de acest tip?
- **De ce** sunt utili constructorii intr-o clasa abstracta daca nu pot fi initializate obiecte de acest tip?

Bonus: Interfete

- Interfata este clasa care contine doar metode virtuale pure.
- Interfetele nu sunt intalnite in limbajul C++, insa pot fi simulate prin clase abstracte.
- Interfetele sunt specific limbajelor evaluate precum Java si C#, limbaje pe care le veti studia in viitor.

Tema

- In clasa Student sa se refaca metoda bugetStudent cum a fost la inceput (sa returneze valoarea 0) si sa se creeze clasa abstracta AStudent care sa contina metoda virtuala bugetStudent().
- Clasa Student mosteneste clasa abstracta AStudent.
- Ce se intampla daca nu suprascriem metoda bugetStudent() in clasa Student?