

TEMA 3

1) Se dau următoarele clase:

```
class Produs:  
    class Review:  
        def __init__(self, autor: str, nota: float):  
            self.autor = autor  
            self.nota = nota  
  
        def __init__(self, nume: str, pret: float):  
            self.nume = nume  
            self.pret = pret  
            self.reviewuri = []  
  
        def adauga_review(self, autor: str, nota: float):  
            self.reviewuri.append(Produs.Review(autor, nota))  
  
        def rating(self):  
            if not self.reviewuri:  
                return 0  
            total = sum(r.nota for r in self.reviewuri)  
            return round(total / len(self.reviewuri), 2)
```

2) Se cer următoarele supraîncărcări:

1. __eq__ : Două produse sunt egale dacă au același nume și același preț.
2. __ne__ : Supraincarcă operatorul pentru inegalitate, folosindu-l pe __eq__.
3. __lt__ și __le__ : Compara două produse după rating.
4. __gt__ și __ge__ : Compara două produse după rating.
5. __add__ : Permite combinarea a două produse pentru a forma un 'pachet' (nume combinat + reviewurile celor două).
6. __iadd__ : Permite adăugarea unui review prin sintaxa `produs += (autor, nota)`.
7. __mul__ : Permite scalarea prețului (`produs * factor` → produs nou cu prețul ajustat).
8. __sub__ : Elimină toate reviewurile unui anumit autor (`produs - autor`).
9. __getitem__ : Permite accesarea reviewurilor prin index.
10. __setitem__ : Permite modificarea notei unui review prin `produs[index] = nota`.

11. `_contains_` : Verifică dacă un autor a dat review (autor in produs).
12. `_len_` : Returnează numărul de reviewuri.
13. `_call_` : Apelează produsul ca o funcție și returnează rating-ul.
14. `_bool_` : Returnează True dacă produsul are măcar un review.
15. `_and_` : Returnează autorii comuni care au dat review la produse.
16. `_or_` : Creează un produs nou care păstrează pentru fiecare autor reviewul cu nota cea mai mare.
17. `_radd_` : Permite sumarea produselor într-o listă, returnând media ratingurilor.
18. `_iter_` : Permite iterarea prin reviewuri (ex: for autor, nota in produs).

3) Testare în main (pentru toate supraîncărările) se face pentru cel puțin 2 produse cu minim 5 review-uri fiecare