

Să se scrie o aplicație care modelează un website de imobiliare. Aplicația conține Case și Apartamente, care au atât atribute comune pentru orice Proprietate, cât și atribute specifice.

Se citesc de la tastatură, pe linii diferite, următoarele informații:

- a. Numărul de proprietăți care vor fi introduse în aplicație
- b. Pentru fiecare proprietate:
  - Un caracter, reprezentând tipul proprietății ('c' pentru casă sau 'a' pentru apartament):
  - Un număr întreg reprezentând prețul proprietății
  - Un std::string (un singur cuvânt), reprezentând locația proprietății
  - Un număr întreg, reprezentând suprafața construită, în mp (facem presupunerea că pentru o casă, toate etajele au aceeași suprafață)
  - Suplimentar, pentru fiecare tip de proprietate se citesc separat următoarele date numere întregi:
    - Casă: Numărul de etaje (incluzând parterul)
    - Apartament: Dimensiunea terasei
- c. O valoare întreagă în intervalul [1, 4] în funcție de care se realizează una dintre operațiile prezentate mai jos:

1. (x1) **Afișarea proprietăților.** Pentru fiecare proprietate se vor afișa pe ecran, în ordinea în care au fost citite de la tastatură următoarele date, sub forma:

- Pentru casă: Casa <pret> <locatie> <suprafata\_construita> <numar\_etaje>
- Pentru apartament: Apartament <pret> <locatie> <suprafata\_construita> <dimensiune\_terasa>

2. (x2) **Afișarea suprafeței totale.** Să se afișeze, pentru fiecare proprietate, în ordinea citirii de la tastatură, suprafața totală. Acesta se calculează în felul următor:

- Pentru casă: suprafața construită \* număr\_etaje
- Pentru apartament: suprafața construită + dimensiune\_terasă

3. (x1) **Afișarea prețului/mp.** Să se afișeze, pentru fiecare proprietate, în ordinea citirii de la tastatură, prețul/mp. Acesta se calculează în felul următor: preț / suprafața\_totală. Valorile se afișează cu 2 zecimale.

4. (x2) **Cea mai bună ofertă.** Se dorește obținerea celor mai bune oferte pentru proprietăți și afișarea lor în ordine **crescătoare**, în funcție de prețul/mp calculat la subpunctul 3. **Dacă 2 sau mai multe proprietăți au prețul/mp egal, se va afișa mai întâi proprietatea cu preț mai mic.**

Pentru fiecare proprietate, se va afișa pe un rând descrierea similar cu subpunctul 1, iar pe următorul valoarea prețului/mp, calculată și afișată similar subpunctului 3.

5. (x2) **Filtrare.** Se citesc de la tastatură atribute care vor fi folosite pentru filtrarea proprietăților, **în ordinea în care acestea au fost citite și pentru datele de la intrare.** Dacă nu se dorește filtrarea după un atribut, acesta va avea valoarea 0. Dacă atributul are o valoare numerică diferită de 0, aceasta va reprezenta valoarea minimă pentru acel atribut. Dacă atributul este de tip std::string și are o valoare diferită de 0, se va filtra după valoarea exactă.

Să se afișeze, similar subpunctului 1, proprietățile care au atributele descrise de filtru.

Observații:

1. **Se vor crea minim 3 clase: Casa, Apartament, care moștenesc clasa abstractă Proprietate**
2. Se vor respecta principiile încapsulării și abstractizării - rezolvările strict procedurale nu se vor lua în considerare

1. Afişarea proprietăţilor

Input	Output
3 c 250000 Bucuresti 110 2 c 40000 Blejo 40 1 a 170000 Bucuresti 70 25 1	Casa 250000 Bucuresti 110 2 Casa 40000 Blejoi 40 1 Apartament 170000 Bucuresti 70 25

2. Afişarea suprafeţei totale

Input	Output
4 c 250000 Bucuresti 110 2 c 40000 Blejo 40 1 a 170000 Bucuresti 70 25 a 100000 Ploiesti 100 5 2	220 40 95 105

3. Afişarea suprafeţei totale v2

Input	Output
2 c 250000 Bucuresti 95 3 c 400000 Blejo 500 1 2	285 500

4. Afişarea preţului/mp

Input	Output
4	1136.36
c	1000.00
250000	1789.47
Bucuresti	952.38
110	
2	
c	
40000	
Blejo	
40	
1	
a	
170000	
Bucuresti	
70	
25	
a	
100000	
Ploiesti	
100	
5	
3	

5. Cea mai bună ofertă

Input	Output
4	Apartament 100000 Ploiesti 100 5
c	952.38
250000	Casa 40000 Blejoi 40 1
Bucuresti	1000.00
110	Casa 250000 Bucuresti 110 2
2	1136.36
c	Apartament 170000 Bucuresti 70 25
40000	1789.47
Blejoi	
40	
1	
a	
170000	
Bucuresti	
70	
25	
a	
100000	
Ploiesti	
100	
5	
4	

6. Cea mai bună ofertă v2

Input	Output
6 c 250000 Bucuresti 110 2 c 40000 Blejo 40 1 a 170000 Bucuresti 70 25 a 100000 Ploiesti 100 5 a 50000 Craiova 45 5 c 150000 Sinaia 75 2 4	Apartament 100000 Ploiesti 100 5 952.38 Casa 40000 Blejoi 40 1 1000.00 Apartament 50000 Craiova 45 5 1000.00 Casa 150000 Sinaia 75 2 1000.00 Casa 250000 Bucuresti 110 2 1136.36 Apartament 170000 Bucuresti 70 25 1789.47

7. Filtrare v1

Input	Output
3 c 250000 Bucuresti 110 2 c 40000 Blejo 40 1 a 180000 Bucuresti 70 25 5 c 0 0 0 0	Casa 250000 Bucuresti 110 2 Casa 40000 Blejoi 40 1
Filtrele încep după ce se citesc cele 3 proprietăți. Singura valoare diferită de 0 este 'c', reprezentând un filtru pentru case. Deci, se vor afișa doar cele 2 case.	

8. Filtrare v2

Input	Output
6 c 250000 Bucuresti 110 2 c 40000 Blejoii 40 1 a 170000 Bucuresti 70 25 a 100000 Ploiesti 100 5 a 50000 Craiova 45 5 c 150000 Sinaia 75 2 5 a 90000 Bucuresti 60 0	Apartament 170000 Bucuresti 70 25
Filtrul caută apartamente cu preț peste 90000, în Bucuresti si cu suprafata construită peste 60mp. Pentru dimensiunea balconului valoarea este 0, deci nu contează. O singură proprietate se potrivește filtrării.	