

Evaluare – Test 2

Timp de lucru: 1 oră și 40 minute

Încărcați aici rezolvarea, cu denumirea **<cod>.zip**, unde <cod> reprezintă codul vostru din lista cu note, până la expirarea timpului de lucru: https://ubbcluj-my.sharepoint.com/:f/g/personal/vlad_ionescu_ubbcluj_ro/EnTCUD5PcyBKmA_8WCpMPdoB6Orv1lzPR2hOCiOLHX0QUA

Scrieți un program cu meniu de tip consolă. Programul va reține datele într-un fișier. Vor fi suportate următoarele funcționalități:

1. [1p] Adăugare localitate: id, nume (string nenul), tip (sat, oraș, minicipiu).
2. [2p] Adăugare rută autocar: id, id_oraș_pornire, id_oraș_oprire (trebuie să existe și să fie distincte), preț, dus-întors (true / false).
3. [2p] Afișarea localităților ordonate crescător după numărul de rute dus-întors care pornesc din ele. Se va afișa și acest număr.
4. [2p] Afișarea tuturor rutelor care se opresc într-o localitate minicipiu. Se vor afișa și denumirile localităților între care e definită ruta.
5. [3p] Export JSON: se creează un fișier JSON cu un nume citit de la tastatură care va conține localitățile, iar în cadrul fiecărui obiect X care reprezintă o localitate, va exista o listă cu numele localităților în care ajunge cel puțin o rută care pornește din X.

De exemplu, pentru localitățile *oradea*, *cluj-napoca*, *chinteni* și rutele *oradea->cluj-napoca*, *oradea->chinteni*, *cluj-napoca->chinteni*, *chinteni->cluj-napoca*, un export valid ar fi:

```
{
  "oradea": ["cluj-napoca", "chinteni"],
  "chinteni": ["cluj-napoca"],
  "cluj-napoca": ["chinteni"]
}
```

Punctajul pe fiecare cerință se acordă astfel:

- 25% corectitudinea implementării.
- 25% arhitectură stratificată și interfață utilizator user friendly.
- 25% specificații scrise corect (unde se aplică) și denumiri sugestive.
- 25% teste relevante și scrise corect (unde se poate).

O cerință nefuncțională se notează cu 0. Nu se acordă punctaje parțiale în afară de cele rezultate din baremul de mai sus.