

Elementele definite vor fi punctate doar dacă sunt complete și au fost testate în main().

Evaluare este orală și soluția este luată în considerare dacă autorul poate explica/justifica implementarea și dacă poate face modificări solicitate în timpul evaluării.

Examen 30 iunie 2020 ora 14.00

Să se scrie o aplicație orientată obiect pentru **evidenta analizelor medicale gestionate de un laborator**; este necesară cunoașterea numelui pacientului, a vârstei, a datei nașterii (stocată ca vector dinamic de numere) și a costului.

Pentru domeniul dat, să se definească o clasă cu membri de tip **public**, **private**, **protected**, **const**, **static**, care să conțină:

(1p) definirea corectă a clasei;

(2p) constructor cu parametri, constructor fără parametri, constructor de copiere, destructor și **operator<<** pentru afișare (0.5 puncte/funcție/operator) ;

(2p) **operator=**, un operator matematic binar, operatorul cast și încă doi operatori adecvați domeniului (de comparare, aritmetici sau indexare sau de alt tip) (0.5 puncte/funcție/operator) ;

(1p) două funcții de tip accesori (dintre care cel puțin o metodă set care să valideze datele de intrare) și două metode proprii clasei (0.25 puncte/funcție/operator);

(1p) funcții sau operatori pentru salvarea și restaurarea obiectelor în/din fișiere **binare**, permanente.

(2p) Definiți încă **o clasă, care este într-o relație de tip „is a”** cu prima clasă (noua clasă adaugă cel puțin 2 atribute); exemplificați conceptul de supradefinirea + funcții virtuale și în main() definiți un vector dinamic care să conțină adrese de obiecte de ambele tipuri (exemplificați avantajele virtualizării).

(1p) Transformați una din clase într-o clasă template sau instanțiați o clasă template STL, pentru domeniul dat.

