

Cerinte de zi - POO / OOP

Lab2-4-Iteratia 2:

- Acoperire 100% la teste
- Sa compileze fara warninguri cu warning level 4. In VS: Click dr. proiect -> Properties -> C/C++ -> General: Warning Level = W4 Treat Warnings As Errors = Yes
- Sa nu aiba memory leak. Folositi CRT Library (gasisti in cursul 3 informatiile)
- Modificati in proiect functia de sortare. Sa fie o singura metoda generala de sortare care primeste ca parametru o functie de comparare

Lab5:

- Acoperire $\geq 99\%$ la teste
- Sa compileze fara warninguri cu warning level 4. In VS: Click dr. proiect -> Properties -> C/C++ -> General: Warning Level = W4 Treat Warnings As Errors = Yes
- Sa nu aiba memory leak. Folositi CRT Library (gasisti in cursul 3 informatiile)
- Implementati un nou algoritm de sortare (daca colegul vostru a implementat in proiectul primit bubble sort, voi puteti face selection sort sau insertion sort sau altceva). Adaugati o optiune in meniu ca sa putem alege algoritmul de sortare dorit.

Lab6-7-Iteratia1:

- Instalat/activat C++ Core Guideline Checker (vezi curs) si reparat (mai) toate erorile raportate.
- Implementati constructorul de copiere de la clasa din domeniu, in corpul constructorului puneti un cout sau println (c++ 23). Apoi rulati aplicatia si vedeti cate obiecte se creeaza si daca sunt prea multe sa le reduceti. La print nu ar trebui sa aveti

Lab6-7-It2:

- code coverage 100%
- no warnings(w4)
- no raw pointers(doar la lista aveti voie)
- no memory leaks
- activati CPP Guideline Checker-rezolvati tot ce tine de const si pointeri
- creati iterator
- adaugati lista/vectorul vostru in proiectul [VectorDinamicCPP.zip](#)(files)- trebuie sa treaca testele de acolo

Lab8-9-It1:

- no memory leak
- code coverage 99%
- fara pointeri
- fara copii inutile

- folositi un dictionar (map, unordered_map) pt crearea unui raport. De exemplu: tip produs <-> cate produse de acest tip exista. Sa fie un raport specific problemei/entitatii

Lab8-9-It2:

- no memory leak
- code coverage 99%
- no warnings(w4)
- creati o clasa pur abstracta Repo si un nou repo. Aplicatia trebuie sa porneasca si cu acest repo, pe langa cel pe care il aveti (pe rand). Repo nou sa tina datele intr-un dictionar (map, unordered_map). La constructor sa primeasca o probabilitate (o valoarea intre 0 si 1) si daca apelam orice metoda din repo, metoda sa arunce exceptie cu probabilitatea data.

Lab10-11-It1:

- code coverage 99%
- la deschiderea ferestrei pe partea dreapta sa apara butoane pentru fiecare tip de produs existent in aplicatie (produs este un exemplu, fiecare foloseste ce entitate are, puteti folosi raportul facut de voi la alt lab). La deschiderea ferestrei se adauga dinamic butoane in functie de ce produse exista deja in aplicatie (similar la celelalte enunturi, ex la filme sa fie buton pentru fiecare gen existent, etc.). La apasarea butonului sa se deschida un MessageBox in care sa apara numarul de produse de tipul respectiv.

Lab10-11-It2:

- Daca aveti in fereastra lista, inlocuiti-o cu un tabel
- Daca aveti in fereastra tabel, adaugati langa o lista

Lab 13-14: NU EXISTA CERINTA