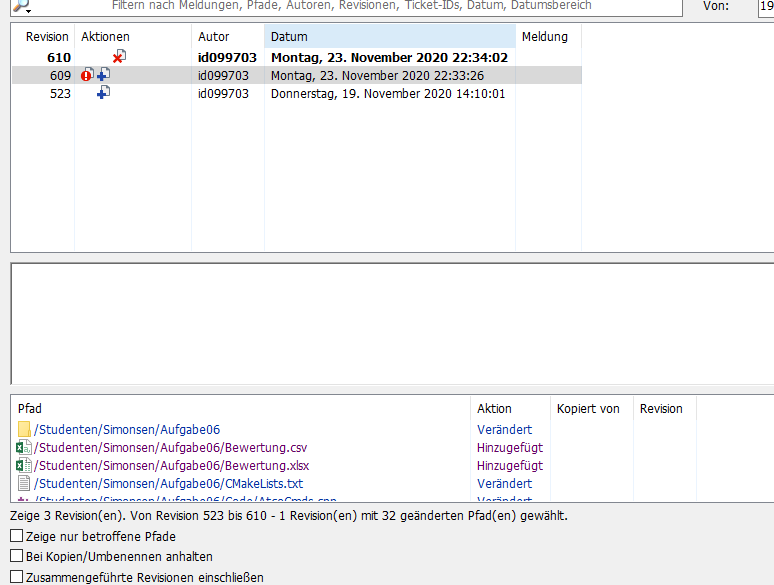
Feedback auf Basis von Rev 610 am 27.11.2020



Code ist nicht Compilierbar und enthält Warnungen auf Stufe 3 und auf Stufe 4

1>------ Neues Erstellen gestartet: Projekt: DynVektorAlsKlasse-Loes, Konfiguration: Debug x64 ------

1>AtcoCmds.cpp

1>AtcoCommand.cpp

1>R:\Studenten\Simonsen\Aufgabe06\Code\AtcoCmds.cpp(73,34): warning C4267: "=": Konvertierung von "size\_t" nach "int", Datenverlust möglich

1>R:\Studenten\Simonsen\Aufgabe06\Code\AtcoCommand.cpp(98,32): warning C4267: "=": Konvertierung von "size\_t" nach "int", Datenverlust möglich

1>R:\Studenten\Simonsen\Aufgabe06\Code\AtcoCommand.cpp(123,54): warning C4458: Deklaration von "wordSequence" blendet Klassenmember aus

1>FileName.cpp

1>R:\Studenten\Simonsen\Aufgabe06\Code\AtcoCommand.h(42,17): message : Siehe Deklaration von "AtcoCommand::wordSequence"

1>R:\Studenten\Simonsen\Aufgabe06\Code\FileName.cpp(23,35): warning C4101: "e": Unreferenzierte lokale Variable

1>R:\Studenten\Simonsen\Aufgabe06\Code\FileName.cpp(29,43): warning C4458: Deklaration von "name" blendet Klassenmember aus

1>main.cpp

1>R:\Studenten\Simonsen\Aufgabe06\Code\FileName.h(24,17): message : Siehe Deklaration von "FileName::name"

1>R:\Studenten\Simonsen\Aufgabe06\Code\dosFarben.h(34,1): warning C4005: "\_MSC\_VER": Makro-Neudefinition

1>R:\Studenten\Simonsen\Aufgabe06\Code\main.cpp : message : Siehe vorherige Definition von "\_MSC\_VER"

1>Processing.cpp

1>Tests.cpp

1>C:\Program Files (x86)\Windows Kits\10\Include\10.0.18362.0\shared\winapifamily.h(19,5): fatal error C1017: Ungültiger Ausdruck für Ganzzahlkonstante

1>R:\Studenten\Simonsen\Aufgabe06\Code\Tests.cpp(37,41): warning C4101: "e": Unreferenzierte lokale Variable

1>R:\Studenten\Simonsen\Aufgabe06\Code\Tests.cpp(189,118): warning C4804: ">": unsichere Verwendung des Typs "bool" in einer Operation

1>R:\Studenten\Simonsen\Aufgabe06\Code\Tests.cpp(278,18): warning C4305: "return": Verkürzung von "int" in "bool"

1>Util.cpp

1>Code wird generiert...

1>R:\Studenten\Simonsen\Aufgabe06\Code\Tests.cpp(299,18): warning C4305: "return": Verkürzung von "int" in "bool"

1>Die Erstellung des Projekts "DynVektorAlsKlasse-Loes.vcxproj" ist abgeschlossen -- FEHLER.

========== Alles neu erstellen: 0 erfolgreich, 1 fehlerhaft, 0 übersprungen ==========

1>main.cpp

1>R:\Studenten\Simonsen\Aufgabe06\Code\dosFarben.h(34,1): warning C4005: "\_MSC\_VER": Makro-Neudefinition

1>R:\Studenten\Simonsen\Aufgabe06\Code\main.cpp : message : Siehe vorherige Definition von "\_MSC\_VER"

#include <iostream>

#define \_MSC\_VER = 1900;

#if (!defined(\_MSC\_VER))

Das #if ist ja immer falsch.

Weitere Bewertung auf Basis von

#include <iostream>

#if (!defined(\_MSC\_VER))

#define \_MSC\_VER = 1900;

#endif

#if (!defined(\_MSC\_VER))

static const short RED\_SCREEN\_COLOR = 0;

Die Lösung mit chrono und damit garantieren, dass es eine Minute testet findet ich ganz gut.

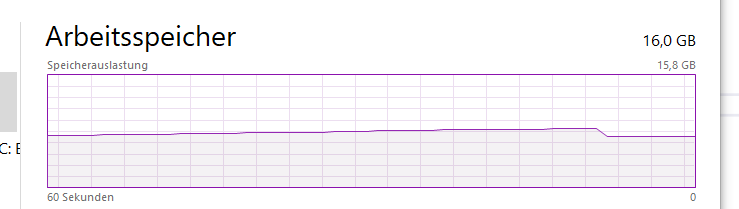
Geht aber bei mir nicht.

Ich hatte vergessen auch das return auszukommentieren.

Man sollte aber auch die Ausgaben auskommentieren. Das frist sonst die meiste Zeit.

Erst dann kann man hinreichend oft die Tests ausführen.

Man erkennt dann auch, dass der Speicherbedarf ansteigt oder eben nicht



Bei Ihnen ist das nicht Fall. Ich habe das verhalten künstlich erzeugt, indem ich Ihr Free um das delete „befreit“ habe.

std::string \*GetCommands() const;

Ganz böse Funktion. Sie nennen die const, was auch richtig ist, Sie ändern (noch) kein Attribut damit, aber Sie geben einen Zeiger auf die Interna zurück.

Eigentlich sollte Sie also

const std::string \*GetCommands() const;

Sie geben aber gerade keinen konstanten Zeiger zurück, sondern ändern über den Zeiger etwas

std::string \*commands = atcoCommand.GetCommands();

commands[0] = "test";

Die Methode also nicht GetX nennen, höchstens

std::string \*GetCommandsForModication() const;

oder besser

void SetCommands(std::string text, no i) {

commands[i] = text;}

Insgesamt macht der Code einen guten Eindruck. Sie kommentieren nicht „übermäßig“, die Funktionen?