# **2. beadandó Molnár Dániel – BMA0B9**

Az Országos Horgász Bajnokság mindegyik versenyén feljegyezték a versenyzők eredményeit, és egy szöveges állományban rögzítették az adatokat. Ennek minden sora tartalmazza a versenyző horgász és a verseny azonosítóját (szóközök nélküli sztring), majd halfajta-méret párok (szóközök nélküli sztring és természetes szám párok) formájában az adott versenyen elért fogásokat. Az adatok szóközökkel vagytabulátorjelekkel vannak egy soron belül elválasztva. A szöveges állomány sorait versenyek szerint rendezték. Feltehetjük, hogy a szöveges állomány helyesen van kitöltve.

Példa az állomány egy sorára:

JANIBÁ Kiliti0512 Ponty 45 Ponty 53 Harcsa 96

(1) Ki, és melyik versenyen fogott keszeget? Soroljuk fel, a megfelelőket!

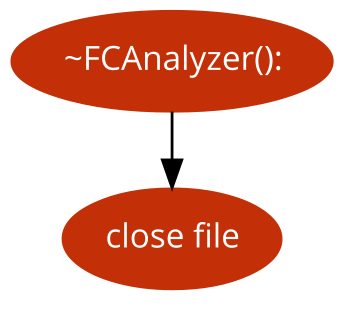
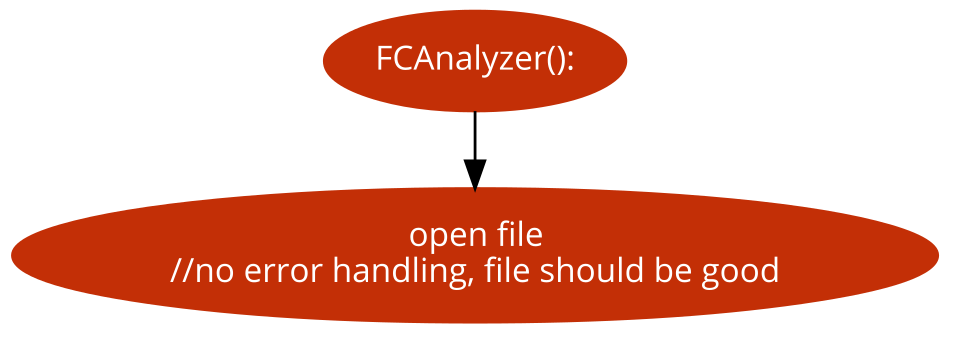
(2) Volt-e olyan verseny, ahol keszegből fogták a legnagyobb halat?

Specifikáció:

A képen képernyőkép látható

Automatikusan generált leírás

FCAnalyzer: Ctor and Dtor



parseData():A képen képernyőkép látható

Automatikusan generált leírás

upsert() (update and insert):

A képen narancs, fénykép, különböző, fekete látható

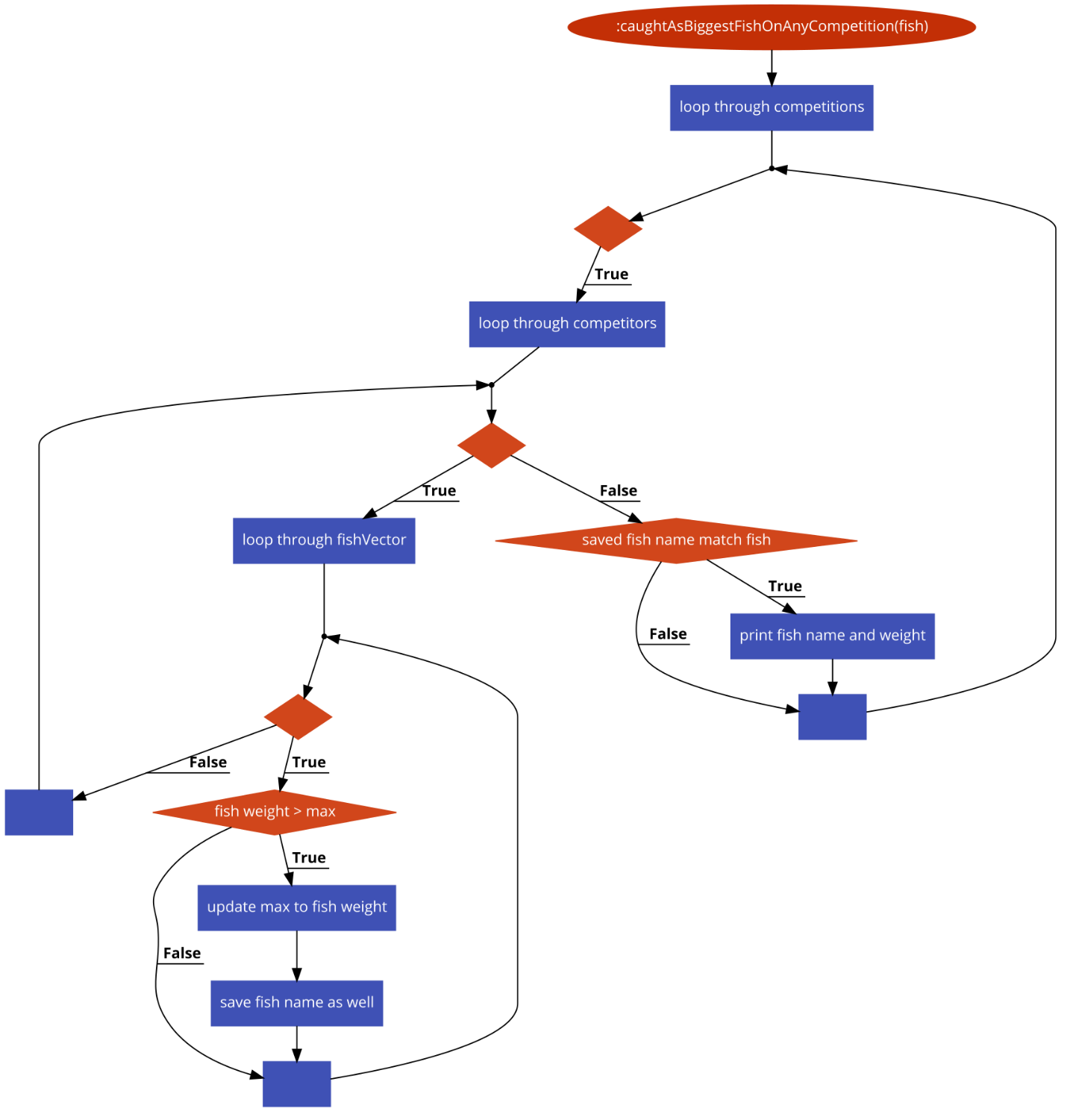
Automatikusan generált leírás

whoCaughtTheFish():

A képen képernyőkép látható

Automatikusan generált leírás

caughtAsBiggestFishOnAnyCompetition():



App:

App(const std::string& filename) : fcaPtr(std::make\_unique<FCAnalyzer>(filename)) {}

processData():

parsing data + printing values if available

Catch test framework added as well, binary under: /cmake-build-debug/Catch\_tests/CMakeFiles/Catch\_tests\_run.exe

Program binary: /cmake-build-debug/fishing\_competition\_analyzer.exe

Tests:

1. Functionality testing:
   1. file should be ok so no tests and exception handling for that
   2. check parsing – each data should be in the right place
   3. check the 2 functions:
      1. whoCaughtTheFishTest – check if the returned vector contains the right value
      2. caughtAsBiggestFishOnAnyCompetitionTest – check if t he returned value is the right value