

Eötvös Loránd Tudományegyetem
Fizika BSc, TTK

Csillagászati kalkulátor C++ és Pythonban történő megvalósítása

Pál Balázs

Készítve az ELTE Fizika BSc, informatikus-fizikus szakirány,
"A Kutatómunka információs eszközei" című tárgy gyakorlati
beadandójaként, 2017. május

Abstract

Az egyszerűbb csillagászati számítások is már sok, monoton számítási műveletet igényelnek. Ezek többségében gömbháromszögekben történő geometriai számolások, vagy teljesen empirikus módon definiált összefüggések szükséges kombinációiból állnak. A közeli égbolton látható mozgások (Nap, bolygók, holdak, stb.) látszólagos, vagy valódi, valamint a látható távolabbi objektumok látszó mozgásainak modelljei emiatt könnyen programozható rendszert alkotnak. A beadandó feladatban különböző csillagászati számításokhoz írtam egy terminálban futó, ASCII grafikával rendelkező segédprogramot.

Tartalom

Abstract	i
1 Felkészülés	1
1.1 A csapat	1
1.2 Motiváció	1
2 Programozás	2
2.1 Használt programok	2
2.2 A megírás folyamata és a program szerkezete	2
2.3 Alkalmazás	3
3 Összegzés	5
3.1 Jövőbeli tervek	5
3.2 Kitekintés	5

1. Felkészülés

1.1 A csapat

A csapatom többi tagja sajnos feladta az órát és nem is értem el őket, mikor elkezdtem dolgozni a beadandón, így egyedül maradtam. Ennek ellenére nagyrészt - amennyire az időmből telt - befejeztem a kitűzött célokat.

1.2 Motiváció

Az idei félévben felvettem egy "Csillagászati észlelési gyakorlatok" nevű tárgyat, amin különféle csillagászati problémákkal és azok megoldásaival ismerkedtünk meg. Az egyik évvégi beadandó egy napóra árnyékának leprogramozása volt, több írásbeli feladattal egyetemben. Úgy döntöttem, hogy a kettő órát összekapcsolva egy részletes kalkulátort készítek - aminek egy része pl. a napóra is -, aminek a segítségével rengeteg órán is tanult problémára könnyen és általánosan választ tudok majd kapni.

2. Programozás

2.1 Használt programok

A teljes programkódot Windows 7, valamint Windows 10 alatt írtam, legnagyobb része VS Code-ban és VS 2015 Community-ban született, és teljesen nulláról lett megírva. Több logikai funkció teszteléséhez is Jupyter Notebookot használtam. Mivel két gépen is programoztam - egyetemen laptopon, otthon asztali gépen - ezért nagyon aktívan hasznát vettem a GitHub-nak is, hogy folyamatosan push-oljam és pull-oljam a projektet a két gép között. Ezt az Anaconda mellé érkező, Git-es MinGW terminálból valósítottam meg, de volt több alkalom, hogy GitHub Desktop-ot használtam.

Ezt a LaTeX file-t a Sharelatex új bétaverziójában, az Overleaf v2-ben írtam meg (csupán kíváncsiságból, próba gyanánt).

A C++ file-t Clang, valamint CMake segítségével fordítottam le.

2.2 A megírás folyamata és a program szerkezete

A programot először Python 3.6.5 alatt írtam meg, amit hosszú munkával be is fejeztem. Ezt a kódot először a [Csillesz-Calculations-KUTINF](#) nevű GitHub repository-m alatt fejlesztettem.

Miután ez minden tervezett funkcionalitással rendelkezett, elkezdtem a programot átfordítani C++ nyelvre. Ezt közben Clang-al teszteltem és ellenőriztem.

Valamivel a C++-re történő fordítás elkezdése után migráltam a teljes repó tartalmát az eredeti csoportom, [Kutinfo_gyakorlat_KA_LM_PB](#) nevű GitHub repository-jába. A további munkákat

és frissítéseket itt folytattam.

Ebből a verzióból az ábrázolások idő hiányában már sajnos kimaradtak, azok csak a Python verzióban vannak meg. Miután a C++ verzió is elkészült - az ábrázolásokat leszámítva - azt CMake-el fordítottam, majd teszteltem le.

A Python verzió jelenlegi, elkészült formájában 4000, míg a C++ verzió valamivel több, mint 6500 soros lett. Mindkét verzióban kihasználtam a nyelv által kínált lehetőségeket, mint pl Pythonban a dict-eket és az azokkal végezhető műveleteket, C++-ben pedig pl. az `std::map` vagy a `try except` struktúrákat és jópár különféle könyvtárat is.

2.3 Alkalmazás

A két program a következő funkciókkal rendelkezik:

- Csillagászati (Égi-) koordinátarendszerek közti átváltás
- Távolságszámítás a Földön két tetszőleges pont között
- Csillagidő számítása tetszőleges helyen és tetszőleges helyi időben
- Napnyugta/Napkelte, polgári-, navigációs-, és csillagászati szürkület, valamint delelés, éjfél és csillagászati éjszaka hosszának és idejének számítása tetszőleges helyen, és időben
- Tetszőleges gömbháromszög hiányzó adatainak kiszámítása az adottak alapján
- A "Csillagászati észlelési gyakorlatok II" című tárgy év végi beadandójának teljes megírása egy gombnyomásra

Kizárólag a Python-os verzió rendelkezik az alábbiakkal:

- Napóra árnyékának felrajzolása az óraszög függvényében a tavaszi- és őszi napéjgyenlőség, a nyári- valamint téli napforduló, valamint opcionálisan egy tetszőleges választott napon az év során, tetszőlegesen választott évben és helyen
- A Nap analemmájának ábrázolása tetszőleges helyen és évben

A program a Csillagászati észlelési gyakorlatok órán megtanultaknál jóval pontosabb adatokat ad eredményül a különféle számítások során. Több interneten is elérhető komoly szoftverrel, valamint tényleges előrejelzéssel összehasonlítottam az eredményeit, és a pontossága legtöbb esetben 1 percnél is pontosabb.

3. Összegzés

3.1 Jövőbeli tervek

A programomat szeretném további optimalizációkkal és funkciókkal bővíteni, azonban ez majd csak a leadási határidő után fog sorra kerülni.

3.2 Kitekintés

Habár a C++ verzióban nem sikerült már az ábrázolásra épülő problémákat megoldani, a Pythonos verzió, funkcióit tekintve 100%-ban működik. Emellett sikerült elsajátítanom a félévben a GitHub és a CMake, valamint a VS Code használatát is.

Mindezeket összevetve az általam kitűzött célt, és a kötelezően megvalósítandó feladat megoldását sikeresnek tekinthetem.