

Софийски университет „Св.Климент Охридски“

Факултет по математика и информатика

Курсов проект по ООП

летен семестър 2020/2021г.

# Проект - Star Wars Universe 0.1

Изработил: Ния Александрова Абрашева

Специалност: Информационни системи

Курс: 1

Факултетен номер: 72033

**Съдържание**

1. Увод
2. Преглед на предметната област
3. Проектиране
4. Реализиране, тестване
5. Заключение
6. Използвана литература
7. **Увод**
   1. Описание и идея на проекта

Идеята на настоящия проект е да пресъздадем вселената от поредицата „Star Wars“ , да можем да запазваме информацията от тази вселена в текстов файл, както и да я извеждаме от такъв, и да правим различни промени по обитаващите планетите джедаи, както и по самите планети. Информацията е представена на потребителя в удобна форма за четене. Приложението се грижи и за обработката на грешки при въвеждането да командите.

* 1. Цел и задачи за разработка

Целта на приложението е да позволи на потребителя да въведе информацията, която желае, за джедаите и планетите. Възможно е да се добавят както планети, така и джедаи. Приложението позволява и извеждането на информация за най-младия или най-силния джедай на дадена планета, както и за най-използвания цвят на светлинния меч по подаден ранк на дадена планета. Съществуват и команди за извеждане на всички джедаи на дадена планета, както и само на определен обитател.

1. **Преглед на предметната област**
   1. Основни дефиниции, концепции и алгоритми

Един от основните компоненти на проекта е класът Menu, където се приемат команди, подадени от потребителя и се изпълняват. В класа има и функции за валидация на данните. Основните функции, които се изпълняват, са:

* *add\_planet* – добавя нова планета.
* *create\_jedi* – създава нов джедай на определена планета и проверява, ако вече съществува такъв на тази или друга планета, не го добавя.
* *removeJedi* – премахва джедай от определена планета.
* *promote\_jedi* – повишава ранга и увеличава силата на определен джедай.
* *demote\_jedi* - повишава ранга и увеличава силата на определен джедай.
* *get\_strongest\_jedi* - извежда най-силния джедай на подадената планета.
* *get\_youngest\_jedi* – извежда най-младия джедай на подадена планета и е със съответен ранг.
* *get\_most\_used\_saber\_color* - връща най-разпространения цвят на светлинния меч в подадения ранг на съответната планета или връща най-разпространения цвят на светлинния меч в планета, който се ползва от поне един **GRAND\_MASTER.**
* *print* - извежда по подходящ начин името на планетата и населяващите я джедаи или извежда по подходящ начин информацията за джедаят, както и коя планета населява в момента.
* *+* - извежда на екрана в сортиран вид (лексикографски) информацията за населяващите двете планети джедаи.
  1. Основни проблеми, решения и определяне на сложност на задачата

Първият основен проблем е начина, по който ще бъде записана информацията във файл, от който зависи и четенето му. За целта във файла информацията за всеки джедай се записва на нов ред, а преди името на всяка планета се поставя „-“, за да може да бъдат разграничени отделните планети.

Вторият проблем е правилната архитектура на проекта и разпределянето на функциите по класове.

Трети основен проблем е обработката на грешно въведени данни от потребителя относно всяка от командите. Това включва имената на планетите и джедаите, годините, ранга и силата. За целта в суперкласа *Menu* са направени функции, които да проверяват коректността на въведената информация.

1. **Проектиране**

3.1 Обща архитектура

Основните класове, съставящи приложението са:

* *Vector* – пресъздава част от функционалността на вградения контейнер в C++; използва се в класовете Planet и Universe.
* *String* - пресъздава част от функционалността на вградения контейнер в C++; използва се в класа Menu за обработка на подадените от потребителя команди.
* *Jedi* – представя информацията за даден джедай- име, години, сила, цвят на светлинния меч и ранг.
* *Planet* – представлява вектор от джедаи и всяка планета има име; всички функции, които трябва да изпълнява приложението (без тези за работа с файлове) са развити в този клас.
* *Universe* – представлява вестор от планети; изпълнява функциите от класа Planet на базата на името на планетата (ако не съществува планета с такова име се извежда подходящо съобщение); тук са развити и функциите за работа с файлове.
* *Menu* – суперкласа на приложението; тук се изпълняват всички команди в приложението.

1. **Реализация, тестване**
   1. Тестване

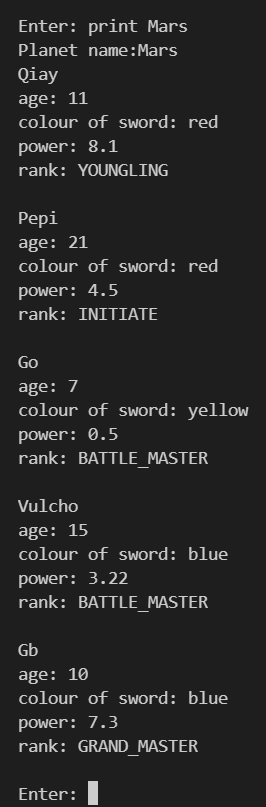
Във файла *tests.cpp* са направени тестове с помощта на библиотеката *doctest.h*

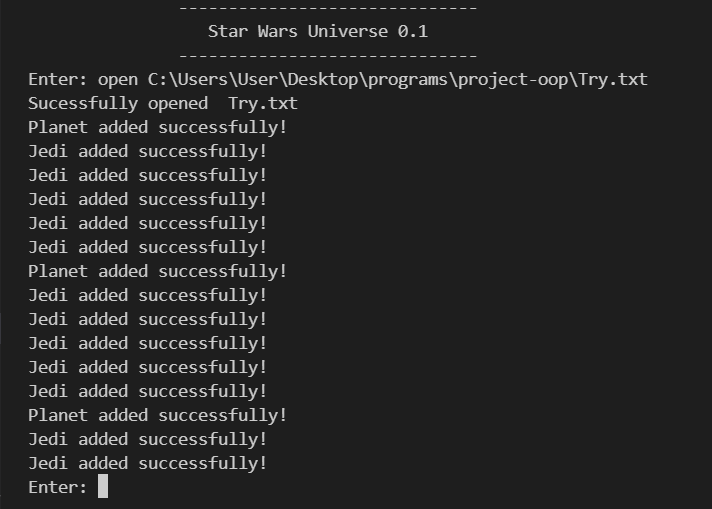
* 1. Управление на паметта

За класовете *Vector* и *String* заделяме необходимото количество динамична памет, като може да изчистваме заделената памет с функцията *clear()* без да унищожаваме обекта от определения клас. Чрез това заделяне (и освобождаване) на динамична памет гарантираме запазването на нужната информация в класовете *Planet* и *Universe* и също постигаме оптималност на кода. В класа *Jedi* използваме динамична памет за името и цвета на светлинния меч.

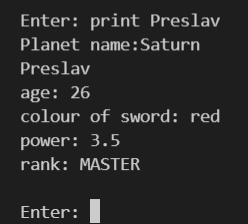
* 1. Примери

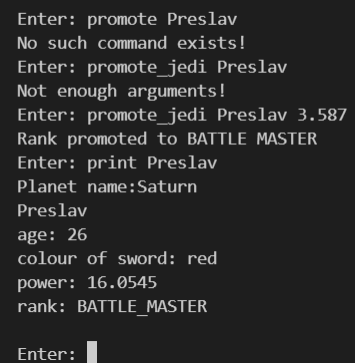
Тук са представени примери за някои от командите.

Фигура 4.3.1 Извеждане на планета, подредена по първи ключ – в нарастващ ред по ранг, и по втори – лексикографски по името

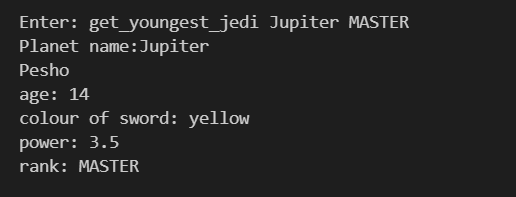


Фигура 4.3.2 Отваряне на съществуващ файл

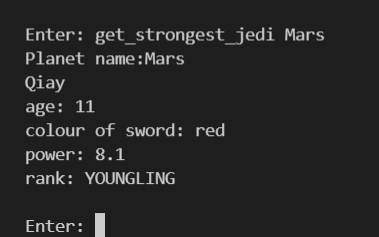
 Фигура 4.3.3 Извеждане на джедай



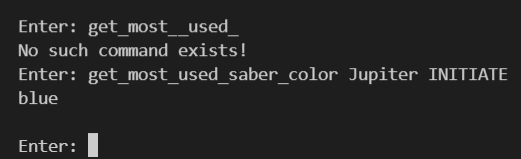
Фигура 4.3.4 Подаване на грешна команда и промотиране на джедай



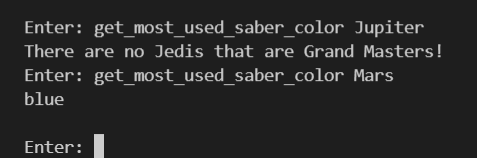
Фигура 4.3.5 Извеждане на най-младия джедай със съответен ранг на определена планета



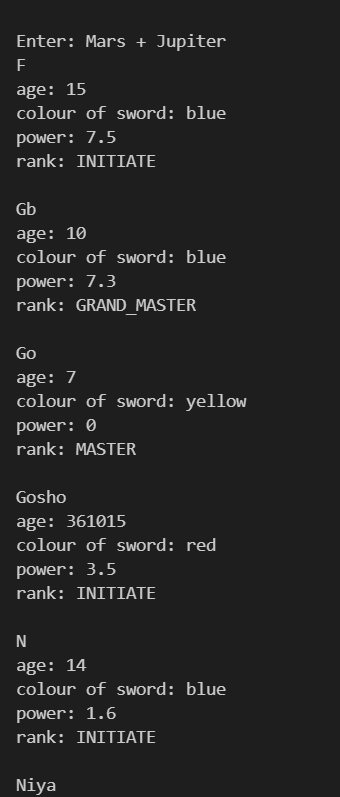
Фигура 4.3.5 Извеждане на най-силния джедай в определна планета



Фигура 4.3.6 Извеждане на най-използвания цвят на светлинен меч в определена планета по подаден ранг



Фигура 4.3.7 Извеждане на най-използвания цвят на светлинен меч в определена планета от поне един *GRAND\_MASTER*



Фигура 4.3.8 Извеждане на джедаите от две планети, подредени лексикографски



Фигура 4.3.9 Излизане от програмата



Фигура 4.3.10 Затваряне на файл без запазване на промените и зареждане на друг

1. **Заключение**

Настоящият проект е примерно представяне на света от хитовата поредица *Star Wars*. Приложението поддържа команди, свързани с въвеждането, извеждането, съхраването и променянето на информация за джедаи, обитаващи конкретна планета. Успешно успява да обработва грешки, направени от потребителя при въвеждане.

**Използвана литература:**

* <https://www.geeksforgeeks.org/>
* <https://stackoverflow.com/>
* <https://www.cplusplus.com/>

**Връзка към хранилището в *Github*:** <https://github.com/nabrasheva/Star-Wars-Universe-1.0>