

Проект „Склад“
по
Обектно-Ориентирано
Програмиране

Изготвил:Йордан Павлов F№71978

Съдържание:

Глава 1. Увод

- 1.1. Описание и идея на проекта
- 1.2. Цел и задачи на разработката
- 1.3. Структура на документацията

Глава 2. Преглед на предметната област

- 2.1. Основни дефиниции, концепции и алгоритми, които ще бъдат използвани
- 2.2. Дефиниране на проблеми и сложност на поставената задача
- 2.3. Подходи, методи (евентуално модели и стандарти) за решаване на поставените проблемите
- 2.4 Потребителски (функционални) изисквания (права, роли, статуси, диаграми, ...) и качествени (нефункционални) изисквания (скалируемост, поддръжка, ...)

Глава 3. Проектиране

- 3.1. Обща архитектура – ООП дизайн
- 3.2. Диаграми (на структура и поведение - по обекти, слоеве с най-важните извадки от кода)

Глава 4. Реализация, тестване

- 4.1. Реализация на класове (включва важни моменти от реализацията на класовете и малки фрагменти от кода)
- 4.2. Управление на паметта и алгоритми. Оптимизации.
- 4.3. Планиране, описание и създаване на тестови сценарии (създаване на примери)

Глава 5. Заключение

- 5.1. Обобщение на изпълнението на началните цели
- 5.2. Насоки за бъдещо развитие и усъвършенстване

Използвана литература

Глава 1.Увод

- 1.1 Проектът има за цел да наподобява складова програма.Тя трябва да има потребителски интерфейс и чрез команди да може лесно да се работи със склада.
- 1.2 Основната цел на програмата е да се използва лесно чрез команди поради тази причина е създаден клас Сторажде който съдържа необходимите команди,а те са именно добавяне на нов продукт, почистване на склада от ненужни вещи с изтекъл срок на годност, изваждане на продукт(количество), показване на всички налични продукти в склада, може да се изведът всички промени извършени в склада в период зададен от потребителя.Също подробна информацията за всеки продукт се съдържа в файл който може да бъде лесно прочетен от потребителя.
- 1.3 Структурата на документацията се извършва по следните стъпки.Първата е да се представяне на належащите проблеми.Втората е търсенето на достатъчно ефективно решение.Документацията завършва с представянето на различни примери и сценарии с цел да се онагледят поведението на програмата в различните сценарии.

Глава 2.Преглед на предметна област

2.1 За целта на проекта ще бъде използван обектно-ориентирана на програмирането.Тя ще позволи поддръжката на програмата да бъде достатъчно лесна и опростена.

2.2 Трябва да бъдат разрешени няколко проблема за реализацията на пълният проект.Основният е свързан с функционалността трябва да се проверяват потребителските данни за валидност при подаване, да се връщат продуктите който потребителят очаква да види и да се запазват продуктите в файл който да остане и след затварянето на програмта. Друг е свързан с потребителския интерфейс като трябва да се следи за безпроблемната работа с програмата.

2.3 Подходът към решаването на вече зададените проблеми е да създам необходимите методи с помощта на който да коректно изпълнение на програмта и извеждане коректни данни за потребителя.

2.4 Има няколко потребителски изисквания за проекта.Те са свързани с лесната и правилна употреба.Функционални изисквания са правилната подредба на склада и правилното класифициране на данните на отделните продукти.Програмата не трябва да натоварва устройството на потребителя с излишни ресурси.Поддръжката трябва да бъде лесна, защото при възникнали проблеми да бъдат лесно открити и отстранени.

Глава 3.Проектиране

3.1 Обектно-ориентираният дизайн се състои в реализацията на клас Storage, Product, Command и Location.Член данните на Product са съответно име на продукта който е променлива от тип char*, срок на годност който е променлива от тип int, дата на продукта в склада която е променлива от тип int, име на производител който е променлива от тип char*, мерна единица на продукта която е променлива от тип char*, количество на продукта който е променлива от тип int, коментар относно продукта който е променлива от тип char* .

Storage съдържа съответно методите за добавяне на нов продукт, почистване на продуктите с изтекъл срок на годност, показване на промените на в склада по дати зададени от потребителя, извеждане на количество от продукт, извеждане на продуктите в конзолата в лесен вид за прочитане от потребителят, записване на продуктите в файл както и има метод за проверяване коректността на данните по контретно дали датата е с формат iso 8601, както и дали мерната единица съществува в противен случай изсиква от потребителят да се въведат нови стойности.

Член данните на Position са съответно секция който е променлива от тип int, рафт който е променлива от тип int, номер на клетката който е променлива от тип int.

Location съдържа съответните методи да проверява дали локацията е свободна и ,ако не записва съответният продукт там както и метод за изчистване на място.

Класът Command съдържа методи за отваряне,затваряне,запазване на файл, както и помощ с необходимите команди за работа с фаелът.

Глава 4. Реализация, тестване

4.1

```
class Storage : public Product {
    friend class Location;
protected:
    int current_number;
    Product** products;
    void remove_el();
public:
    Storage();
    Storage(int current_number);
    Storage(const Storage& other);
    Storage& operator= (const Storage& other);
    ~Storage();
    void add_product(Product &product, int location);
    void clean_odd_products(int _date);
    void log_products(int start_date, const int end_date);
    void remove_product(const char* name_to_remove, int quantity, const char* unit_to_remove);
    void show_products();
    void move(int i);
    void in_to_file(const string &file_path);
    void print_file(const string &file_path);
};

class Product : public Location{
public:
    char* _name ;
    int expiration_date ;
    int date_in_storage ;
    char* name_of_manufactor ;
    char* _unit ;
    int available_quantity;
    char* _comment ;
```

4.2 Използвал съм вектори както и голямата 5-ца (3-ка) за правилното разпределяне на динамичната памет в програмта.

4.3 Когато потребителят въведе :

```
add banan 3052020 3012020 ivan kg 20 testov_product 10101
```

```
add banan 3052020 3012020 ivan kg 20 testov_product 30101
```

```
show
```

програмата изкарва: name of the product is: banan

available quantity of the product is: 40 kg

```
add kiwi 5052020 5032020 ivan mkr 21 testov_product 10510
```

програмата изкарва: it allow only : gr,kg,ml,l,lbs | please enter the valid unit

```
add milk 1234 3042021 ivan l 25 testov_product 10610
```

програмата изкарва: please enter expiration date for milk

```
add desert 4032021 3042021 ivan kg 27 testov_product 60509
```

```
remove desert 26 kg
```

програмата изкарва: you successfully removed 26 left: 1 kg from desert

```
add chocolate 4052021 3052021 ivan kg 20 testov_product 60509
```

програмата изкарва: please enter new location

```
clean 3052020
```

```
log 1012020 5182020
```

Глава 5. Заключение

5.1. Проектът е завършен с изключение на проверката за локация.Бяха намерени оптимални решения за решаването на проблемите който бяха зададени.Интерфейсът е достатъчно функционален и лесен за употреба.

5.2 Може да се добави автоматично почисване на слада, да увеличи капацитетък за поддържани продукти,метод за промяна на данните на продуктите.

Използвана литература:

<http://www.cplusplus.com/>

<https://www.geeksforgeeks.org/> - за сплитването на командтата по думи