

Глава 1: Увод – Проект „Склад“

Поставена задача:

Да се реализира информационна система, която поддържа и обработва информация за наличната стока в склад.

Складът се състои от идентични секции с идентични рафтове.

В продукт се описва от следните характеристики:

- Име на продукта
- Име на производител
- Срок на годност
- Дата на постъпване в склада
- Количество на продукта в съответната пратка/партида

Също така всеки продукт бива доставен и съхраняван в кашон, който се описва от трите си измерения (Приема се, че всички продукти, които са в един кашон, имат един и същ срок на годност):

- Дължина
- Широчина
- Височина.

За продукта се пази и информация за неговото местоположение в склада:

- Номер на секция
- Номер на рафт
- Пореден номер на кашона на рафта

Потребителят може да обработва информацията в системата, използвайки следните команди

Добавяне на продукт	Потребителят може да добавя продукт в склада в диалогов режим: За да добави продукт, трябва да се въведат името на продукта, името на производителя, датата на постъпване на продукта, срока на годност, както и
---------------------	---

	<p>размерите на кашона (широчина, дължина, височина).</p> <p>Системата намира място на продукта в склада.</p> <p>Внимание: Едноименни продукти, пристигнали в различни кашони, се намират на различни места в склада (тоест не прехвърляме стока от един кашон в друг)</p>
Разчистване на всички продукти с изтекъл срок на годност	Когато потребителят въведе дата, системата премахва всички продукти, които са с изтекъл срок на годност спрямо въведената дата.
Извежда направени промени по наличността на продукти в склада за даден период	Потребителят трябва да въведе периода (под формата на начална дата DD MM YYYY и крайна дата DD MM YYYY), за който иска да му бъде направена справка за промените в наличността. Те са добавяне на продукти или премахване на продукти от склада.
Извежда всички налични продукти в склада	Извежда списък от имената на всички продукти в склада и тяхното количество. Като за едноименните отпечатва общото им количество, независимо от местоположението им и срока на годност.
Изваждане на продукт в диалогов режим	<p>Продукт се изважда като бъдат подадени неговото име и съответното количество, което да бъде премахнато.</p> <p>Системата първо премахва тези продукти, чийто срок на годност е най-близо да изтече. В случай, че се потребителят се опитва да извади по-голямо количество, от колкото има в кашона, програмата ще го попита дали иска да вземе недостига от други партиди на същия продукт.</p>

Глава 2: Проектиране

Допълнителни структури от данни, използвани за реализацията на проекта:

- **Клас Vector<>** - динамичен масив, приемащ произволни типове данни (самостоятелно реализиран)

- **Клас Pair<>** - наредена двойка за произволни типове данни (използван код от упражнения, към който са допълнени 2 оператора)
- **Клас String** – използван код от упражнения

Дизайн на проекта:

- клас **Storage** – поддържа информацията за продуктите в склада
 - клас **Section** – поддържа информация за продуктите в секцията
 - клас **Vector<>** - поддържа информация за продуктите на рафт
 - клас **Product** – съдържа основните характеристики на продукт
 - клас **Date** – поддържа валидни дати и реализира работа с дати
 - клас **String**
- класове от допълнителните структури от данни, спомагащи обработката на продукти в склада

Глава 3: Подход и реализация

- **Добавяне на продукт на рафт**
 - **Идея:** За всеки продукт се избира първото свободно място в секция – рафт. Кашоните могат да се завъртат. Ако един кашон успешно премине проверките дали може да бъде поставен да текущия рафт, то се избира това му измерение, което е най-късо. Системата реди кашоните един до друг. Подреждания, в които има кашони един пред друг или един върху друг са **невалидни**, както и не може да има свободни пространства между кашоните.
 - **Реализация:** Неявно се пазят дължините на кашоните, за да може да се прави бърза проверка колко място остава на рафта. Дължината на всеки кашон от рафт е сбора от дължините на всички кашони преди него и неговата собствена.
- **Премахване на продукт от рафт**
 - **Идея:** Намира се съответния продукт, който трябва да бъде премахнат, премахваме го и преместваме всички кашони след него(долепяме ги).

- **Реализация:** Разглеждат се три случая в зависимост от позицията на кашона на реда:
 - **първи:** всички кашони се преместват с дължина равна на дължината на първия кашон
 - **междинен:** неговата точна дължина ще е разликата от двата му съседни кашона. След като намерим точната дължина на кашона, премествахме всички кашони, които са били отдясно на текущия с толкова, колкото е и дължината му.
 - **последен:** премахва се директно
- **Списъци, поддържащи информацията за продуктите в склада**
 - `Vector< Pair<Product, bool> > actionsMade;` - поддържа данните за всички промени, направени в склада
 - Булева стойност 0 – продуктът е бил добавен
 - Булева стойност 1 – продуктът е бил премахнат
 - `Vector<Product> listOfValidProducts;` - поддържа данните за всички налични продукти, които са в срок на годност до последно разчистване. Данните са сортирани по срок на годност
 - `Vector< Pair<MyString, int> > quantityOfProduct;` - поддържа информация за общото количество на едноименните продукти. Списъкът се обновява при изваждане на продукти.

Глава 4: Тестване

Примерен тест:

```
>i Aspirin | bayer | 22 8 2022 | 12 12 2022 | 20 |10 10 10
>i Trahizan| Engelhard | 22 10 2022 | 12 12 2022 | 10 |20 20 20
>i mig400 berlinChemi | 20 10 2022 | 14 12 2022 | 15 |15 15 15
>i mig400 | berlinChemi | 23 10 2022 | 13 12 2022 | 10 | 15 15 15
>exit
```

Name:aspirin Producer:bayer

ValidityDate:12.12.2022 EnterDate:22.8.2022

Section:0 Shelf:0 Id:0 inserted

Name:trahizan Producer:Engelhard

ValidityDate:12.12.2022 EnterDate:22.10.2022

Section:0 Shelf:0 Id:1 inserted

Name:mig400 Producer:berlinChemi

ValidityDate:14.12.2022 EnterDate:20.10.2022

Section:0 Shelf:0 Id:2 inserted

Name:mig400 Producer:berlinChemi

ValidityDate:13.12.2022 EnterDate:23.10.2022

Section:0 Shelf:0 Id:3 inserted

aspirin 20

trahizan 10

mig400 25

>r mig400 5

>r mig400 10

Do you want to get another stock

1

>exit

aspirin 20

trahizan 10

mig400 10

Name:aspirin Producer:bayer

ValidityDate:12.12.2022 EnterDate:22.8.2022

Section:0 Shelf:0 Id:0 inserted

Name:trahizan Producer:Engelhard

ValidityDate:12.12.2022 EnterDate:22.10.2022

Section:0 Shelf:0 Id:1 inserted

Name:mig400 Producer:berlinChemi

ValidityDate:14.12.2022 EnterDate:20.10.2022

Section:0 Shelf:0 Id:2 inserted

Name:mig400 Producer:berlinChemi

ValidityDate:13.12.2022 EnterDate:23.10.2022

Section:0 Shelf:0 Id:3 inserted

Name:mig400 Producer:berlinChemi

ValidityDate:13.12.2022 EnterDate:23.10.2022

Section:0 Shelf:0 Id:3 removed

Name:mig400 Producer:berlinChemi

ValidityDate:14.12.2022 EnterDate:20.10.2022

Section:0 Shelf:0 Id:2 removed

Глава 5: Заключение

Идеи за бъдещи подобрения:

- Да се разполага със списък от продуктите, които трябва да бъдат доставени в склада. По този начин може да се направи валидация и на самите продукти, освен на коректността на входните им данни
- Подобряване на подредбата. Да се допуска да има кашони, които са един зад друг или един пред друг.

Линк към хранилището, съдържащо проекта:

https://github.com/ElitsaY/OOP_Project_1

Изготвено от Елица Тихомирова Йоткова, студент 1 курс, специалност „Компютърни науки“, поток 1, група 3, ФН: 4MI080119