



Софийски университет  
„Св. Климент Охридски“  
Факултет по математика и  
информатика

# ЗАДАЧИ ЗА ПЪРВО ДОМАШНО

курс Обектно-ориентирано програмиране  
за специалност Информатика, Компютърни науки и Информационни системи,  
летен семестър 2019/2020 г.

## Общи изисквания

Решенията трябва да отговарят на следните изисквания и ограничения:

- За съхранение на потребителите и предизвикателствата да се използва минимално количество динамична памет. Всички масиви, използвани в класовете, трябва да са с точна дължина.
- Отделните елементи на системата да са представени с класове
- Следвайте добрите ООП практики за реализация на тези класове
- В реализираните класове трябва да са добавени подходящи методи, които са необходими за правилното решение на задачата, въпреки че може да не са явно посочени в условието.
- Решение, което не се компилира или грубо нарушава принципите на ООП, ще бъде оценено с 0 точки.
- За реализацията не е позволено използването на класа `std::string`, както и на контейнерите от STL.

# #STAYHOME

Докато трае самоизолацията и си седим вкъщи (което се оказва много трудно), решихме да се разнообразим с това да се предизвикваме по различен начин в социалните мрежи и под тага #stayhome заваляха и #pushupschallenge, #toiletpaperchallenge, #качисрамнасимкаchallenge и т.н. Програмистите, така ентусиазирани, че и другите сега споделят живота им на самоизолация, решили да зарадват света с ново приложение “StayHomeChallenges”, което ще е само за предизвикателства. И понеже това била идея на екипа по ООП във ФМИ, а идеята е идея за милиони, те решили да поискат помощ от студентите си. Искаме приложението да достигне до всеки, за това трябва да се разработи както и desktop версия, така и сайт, а най-вече и мобилно приложение. Трябва, разбира се, да се поддържа и от всички операционни системи. Работата е много, затова на студентите е възложена най-тежката задача – да напишат цялото ядро на приложението, на база на което ще работят всички негови версии.

## Бонус {

Понеже навсякъде има интернет пирати, които копират приложения и ги представят за свои, трябва нашето ядро да има предпазна “система” от това да може да се копира.

}

За да може екипът по ООП да започне да разработва поне beta версии на мобилните приложения, настолното приложение и уеб приложението, към студентите е зададено условието да се поддържат следните основни функционалности:

1. Регистриране на потребители
2. Отправяне на предизвикателства

Все още не знаем дали приложението ще пожъне успех и затова не сме сигурни колко потребители ще се регистрират. Поради това приложението трябва да поддържа произволен брой потребители. Всеки потребител трябва да има име и освен това, за да може всеки по света да има възможност да

използва приложението, не е добре да се налага ограничение колко дълго да е то. При регистрация потребителят трябва да въведе основна информация: години (до 90) и мейл адрес— до 100 символа. За да може потребителите с еднакви имена, години и имейл адреси да бъдат различавани, системата трябва сама да слага уникален идентификационен номер при успешна регистрация.

Докато екипът по ООП пише за различните приложения форми за регистрация с полета и бутони, ядрото на системата ще работи с текстови команди. Регистрацията ще става със следната команда:

```
registration Integralcho 19 integral@fmi-sofia.bg
```

Потребителят трябва да може да се регистрира без да се налага да предоставя част от личната си информация като години и имейл адрес:

```
registration Integralcho 19
```

```
registration Integralcho integral@fmi-sofia.bg
```

Разбира се, за да може успешно да бъде предизвикан даден потребител, той трябва задължително да въведе името си.

Следващата функционалност, която трябва да поддържа системата, е отправяне на предизвикателства. Един потребител предизвиква останалите по следния начин:

```
challenge <името на потребителя, който предизвиква> <таг на  
предизвикателство> <имената на предизвиканите, отделени с интервал>
```

Следните проверки трябва да се извършат преди предизвикателството да се регистрира успешно в системата:

- Всеки от потребителите (предизвикващ и предизвикани) трябва да е вече регистриран в системата;
- Тагът задължително започва със знака #. Той идентифицира уникално предизвикателството и има максимална дължина от 31 символа;

- Ако предизвикателството не е било отправяно досега, то се добавя в системата и неговият статус става "new". Ако е било отправено преди, то не се регистрира наново, но се увеличава броя на отправянията му и се сменя статусът му, ако е нужно. Ако е било отправяно между 2 и 10 пъти, статусът му трябва да бъде "quite recently", а от 11 нагоре – "old".

За да знаем дали едно предизвикателство е интересно и успяващо, то има и рейтинг (дробно число), който варира в интервала [-5.0, 10.0]. Екипът по ООП предизвиква студентите (поради дефицит на памет) предизвикателството да се представя с не повече от 40 байта в паметта.

Всеки потребител пази списък от предизвикателствата, които са отправени към него и които все още не е изпълнил. Той изпълнява предизвикателствата със следната команда:

`finish` <таг на предизвикателство> <идентификационен номер на потребителя> <с каква оценка го оценява>

За да разберем какъв е идентификационния номер на потребителя, неговият профил може да се прегледа със следната команда:

`profile_info` <име на потребител>

Като резултат се извежда цялата въведена при регистрация информация за потребителя. Ако някое от полетата не е въведено при регистрация, за негова стойност се изписва "Unknown".

След като предизвикателството е изпълнено, потребителят вече не пази информация за него, а рейтингът му се обновява. Рейтингът се пресмята като средно аритметично от всички дадени оценки. Понеже все още системата е в тестов режим, броят на отправяния на предизвикателство и броят на изпълнения няма да надвишава 65535.

Разбира се, приложението трябва да има и стена, на която може да се виждат всички предизвикателства и да се подреждат по определен начин. Това става със следната команда:

`list_by` <как да са подредени>

като имаме следните възможности за подредба: `newest` (по ред на отправяне, първо най-скорошните), `oldest` (по ред на отправяне, първо най-старите), `most_popular` (по брой отправяния, първо най-често отправяните).

Бонус за спец. Информатика, задължителна част за спец. Компютърни науки:

За да може да изтества по-бързо системата, може да се поддържат и следните команди:

```
load users.txt
```

и

```
load challenges.bin
```

които зареждат в системата съответно потребители (от текстов файл) и предизвикателства (от двоичен файл). Командата `load` трябва автоматично по разширението да определя дали зарежда потребители или предизвикателства.

Пример за изпълнение на програмата (с \$ е отбелязан входът на потребителя):

```
$ registration Integralcho 19 integral@fmi-sofia.bg
```

```
Registration Successful. Please check your email.
```

```
$ registration Mariika 25 qkataMara95@abv.bg
```

```
Registration Successful. Please check your email.
```

```
$ registration DqdoToshko 92 rakiq@gmail.bg
```

```
Registration Fail. You are too old for this App.
```

```
$ registration Milcho 37
```

```
Registration Successful.
```

```
// Следващите примери ще са с потребители, които можем да смятаме, че са вече регистрирани или са прочетени от файла users.txt
```

```
$ challenge Integralcho #toiletpaper Ivan Dragan Marrika
```

```
Integralcho successfully challenged!
```

\$ challenge Milcho izqshlutqchushka Mariika Integralcho

Sorry! Invalid tag of the challenge.

\$ challenge Hristo #reshiSudokuto DqdoToshko

Sorry, some of the users may not be registered!

// Следващите примери ще си представим, че са отправени предизвикателства, които са регистрирани успешно и са отправяни повече от веднъж.

\$ finish #toiletpaper 2 8.6

Well done! May the challenge be with you!

\$ finish #toiletpaper 8 -0.6

Well done! May the challenge be with you!

\$ finish #pushupschallenge 13 7.6

The given id isn't valid. The user with this id do not exist.

\$ finish #pushupschallenge 9 17.6

Sorry. Invalid rating! The challenge is not completed, yet.

\$ profile\_info Integralcho

Searching for users...

1)           Name: Integralcho  
Age: 19  
Email: [integral@fmi-sofia.bg](mailto:integral@fmi-sofia.bg)  
Id: 1

2)           Name: Integralcho  
Age: 37  
Email: Unknown  
Id: 10

3)           Name: Integralcho  
Age: Unknown  
Email: Unknown

Id: 14

\$ profile\_info BabaStanka

Searching for users...

Nothing come out! User with this name do not exist!

\$ list\_by most\_popular

Tag	Rating	Status	Total	Done
#gledaiSerial	10.0	old	1024	1024
#napishiDomashnoPoOOP	-3.5	old	311	258
#HodiBezMaskaBezDaTeGlobqt	4.2	quite recently	8	0
#kachiSnimkaKatoBebe	5.5	new	1	0