

СОФИЙСКИ УНИВЕРСИТЕТ
„СВ. КЛИМЕНТ ОХРИДСКИ“ФАКУЛТЕТ ПО МАТЕМАТИКА
И ИНФОРМАТИКА**ДЪРЖАВЕН ИЗПИТ**
ЗА ПОЛУЧАВАНЕ НА ОКС “БАКАЛАВЪР ПО СОФТУЕРНО ИНЖЕНЕРСТВО”**ЧАСТ I (ПРАКТИЧЕСКИ ЗАДАЧИ)****14.07.2015 г.**

Моля, не пишете в тази таблица!			
Зад. 1		Зад. 5	
Зад. 2		Зад. 6	
Зад. 3		Зад. 7	
Зад. 4		Зад. 8	
Крайна оценка:			

Драги абсолвенти:

- Попълнете факултетния си номер в горния десен ъгъл на всички листа;
- Пишете само на предоставените листове без да ги разкопчавате;
- Ако имате нужда от допълнителен лист, можете да поискате от квесторите;
- Допълнителните листа трябва да се номерират, като номерата продължават тези от настоящия комплект;
- Всеки от допълнителните листа трябва да се надпише най-отгоре с вашия факултетен номер;
- **Решението на една задача трябва да бъде на същия лист, на който е и нейното условие (т.е. може да пишете отпред и отзад на листа със задачата, но не и на лист на друга задача);**
- Ако решението на задачата не се побира в един лист, трябва да поискате нов бял лист от квесторите. В такъв случай отново трябва да започнете своето решение на листа с условието на задачата и в края му да напишете „Продължава на лист № X”, където X е номерът на допълнителния лист, на който е вашето решение;
- Черновите трябва да бъдат маркирани, като най-отгоре на листа напишете „ЧЕРНОВА“;
- На един лист не може да има едновременно и чернова и белова;
- Времето за работа по изпита е 3 часа.

Изпитната комисия ви пожелава успешна работа!

Задача 1 (10т.). Следната задача да се реши на един от езиците за програмиране C++ или Java.

Дадена е `float` матрица `img` с размери $M \leq 10$ реда и $N \leq 10$ стълба. Напишете функция `subsample`, която получава като аргументи M , N и `img` и извежда на екрана матрица `s` с размери $(M+1)/2$ и $(N+1)/2$ (при целочислено деление), всеки елемент `s[i][j]` на която е равен на средно-аритметичното от всички елементи `img[y][x]`, такива че

$$i*2 \leq y \leq i*2+1 \text{ и } j*2 \leq x \leq j*2+1.$$

Например, при матрица `img`, представена таблично по следния начин:

1.0	2.0	3.0
4.5	6.5	7.5

функцията да извежда на екрана:

3.5	5.25
-----	------

Забележка: При избор на Java за език за програмиране е достатъчно да се дефинира статичен метод, който решава задачата.

Задача 2 (10т.). Следната задача да се реши на един от езиките за програмиране C++ или Java.

Да се обозначи явно на кой от двата езика е решавана задачата. При решението на задачата да не се използват библиотеки за работа със структури от данни.

а) Да се дефинира подходяща *индуктивна (рекурсивна)* структура от данни, позволяваща представянето в паметта на програмата на възел на дърво от цели числа (`int`), за което всеки връх може да има произволен брой наследници (0, 1 или повече).

б) Да се дефинира *рекурсивна* функция (или статичен метод)

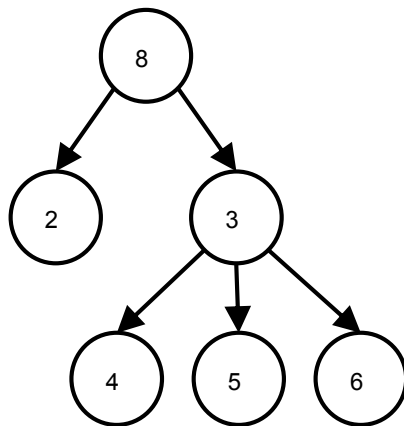
```
[булев тип] member ([подходящ тип] root, int x)
```

чиято стойност е истина точно тогава, когато в дървото с корен, представен от параметъра `root`, съществува възел със стойност `x`.

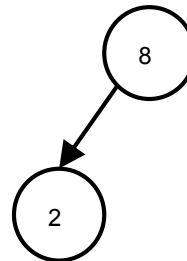
в) Да се дефинира рекурсивна функция (или статичен метод)

```
void filterOdd ([подходящ тип] root)
```

Функцията да премахва (чрез мутация) всяко поддърво t' на дървото с корен, представен от параметъра `root`, за което е изпълнено, че коренът на t' е със стойност нечетно число. На следната фигура е показано примерно дърво преди и след изпълнението на операцията `filterOdd`.



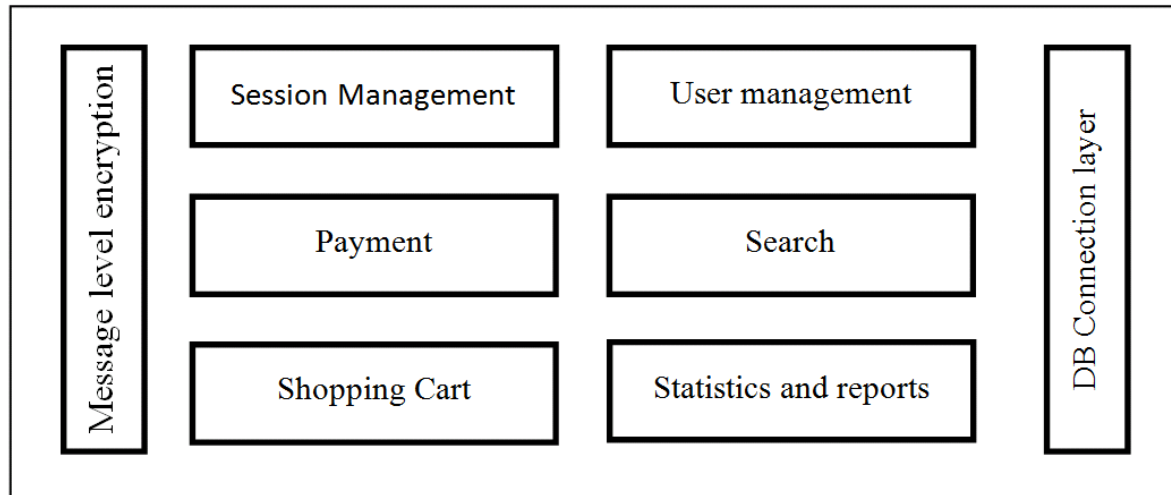
примерно дърво t



дървото t след приложение на `filterOdd`

Забележка: Реализирайте всички конструктори и други операции, които смятате, че са необходими на съответните класове.

Задача 3 (10т.). Налице е следната декомпозиция на модули от високо ниво на система (портал) за онлайн търговия.



Модулите имат следните функционалности:

- Session Management – Управлява активните потребителски сесии и следи политиките за оторизация на потребителите
- XML level encryption – Използва се за допълнително криптиране на потребителските заявки
- Payment – В този модул са реализирани функционалностите за заплащане на стоките, купувани през портала. Осъществява връзката с външни системи за плащане
- Statistics and reports – В този модул е реализирана функционалност по създаване на статистики и справки за най-търсени стоки, най-активни потребители и т.н.
- Shopping Cart – Модул, чрез който се реализира функционалността по заявяване на стоки за купуване, създаване на оферта и издаване на фактури и гаранционни документи.
- Search – Модул, в който са реализирани алгоритмите за търсене на стоки и услуги
- User management – Административен модул за управление на потребителските акаунти
- DB Connection layer – адаптер за връзка с базата данни

Към настоящия момент системата трудно поема пикови натоварвания (например в период на промоции, когато има повишено търсене на дадена стока).

А) Да се напише сценарий за производителност, който да отговаря на изискването системата да може да поеме пикови натоварвания.

Б) Да се напише сценарий за изпитаемост (testability), който да отговаря на изискването системата да дава възможност за максимално покритие на кода чрез минимален брой тестови случаи

В) Да се обнови архитектурата на системата, така че да се удовлетворят двата създадени сценария.

За целта може да се представи декомпозиция на някои модули на по-ниско ниво и/или други структури на системата, по ваша преценка.

Задача 4 (10т.). Дадена е базата от данни Movies.

Таблицата **Studio** съдържа информация за филмови студиа:

name – име, първичен ключ;

address – адрес.

Таблицата **Movie** съдържа информация за филми. Колоните *title* и *year* заедно формират първичния ключ.

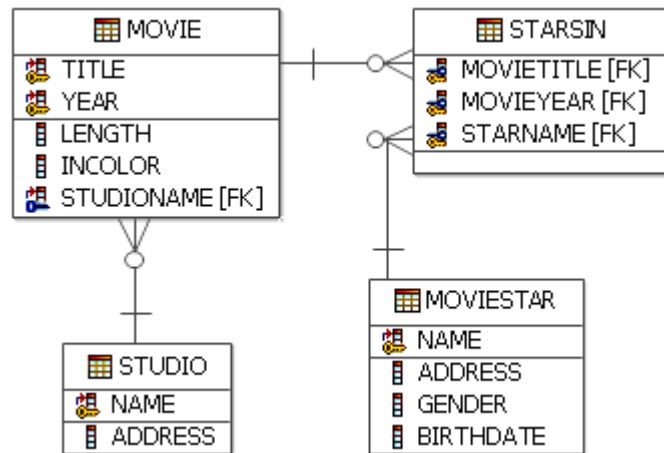
title – заглавие;

year – година, в която филмът е заснет;

length – дължина в минути;

incolor – 'Y' за цветен филм и 'N' за черно-бял;

studioName – име на студио, външен ключ.



Таблицата **MovieStar** съдържа информация за филмови звезди:

name – име;

address – адрес;

gender – пол, 'M' за мъж и 'F' за жена;

birthdate – рождена дата.

Таблицата **StarsIn** съдържа информация за участието на филмовите звезди във филмите. Трите колони заедно формират първичния ключ. Колоните *movietitle* и *movieyear* образуват външен ключ към Movie.

movietitle – заглавие на филма;

movieyear – година на заснемане на филма;

starname – име на филмовата звезда, външен ключ.

1. Да се посочи заявката, която извежда имената на всички филмови звезди, чието име не завършва на "а" и са играли както в цветни, така и в черно-бели филми.

- a) `SELECT name
FROM MovieStar, StarsIn, Movie
WHERE name = starName AND movieTitle = title AND movieYear = year
AND name != '%a' AND inColor = 'y' AND inColor = 'n';`
- б) `SELECT MovieStar.name
FROM MovieStar
WHERE NOT (name LIKE '%a')
AND name IN (SELECT starName
FROM StarsIn
JOIN Movie ON movieTitle = title AND movieYear = year
WHERE inColor = 'y' OR inColor = 'n');`
- в) `SELECT DISTINCT starName
FROM StarsIn
INNER JOIN Movie ON movieTitle = title AND movieYear = year
WHERE starName NOT LIKE '%a' AND inColor = 'y'
AND starName = (SELECT starName
FROM StarsIn, Movie
WHERE inColor = 'n');`
- г) `SELECT starName
FROM StarsIn
JOIN Movie ON movieTitle = title AND movieYear = year
WHERE starName NOT LIKE '%a' AND inColor = 'y'
INTERSECT
SELECT starName
FROM StarsIn
JOIN Movie ON movieTitle = title AND movieYear = year
WHERE inColor = 'n';`

2. Посочете заявката, която извежда за всяка филмова звезда, играла в най-много 5 филма, следната информация:

- име;
- рождена година;
- брой студия, с които е работила.

Ако за дадена звезда няма информация в какви филми е играла, за нея също да се извежда ред (с брой студия, равен на 0).

- a)

```
SELECT DISTINCT name, birthdate.year, COUNT(studioName)
FROM MovieStar, StarsIn, Movie
WHERE name = starname AND
      ((movieTitle = title AND movieYear = year) OR title IS
NULL)
GROUP BY name, birthdate.year
HAVING COUNT(title) <= 5;
```
- б)

```
SELECT name, YEAR(birthdate), COUNT(DISTINCT studioName)
FROM MovieStar
LEFT OUTER JOIN StarsIn ON name = starname
LEFT JOIN Movie ON movieTitle = title AND movieYear = year
GROUP BY name
HAVING COUNT(title) <= 5;
```
- в)

```
SELECT          StarsIn.starname,          YEAR(birthdate),
COUNT(DISTINCT studioName)
FROM Movie
JOIN StarsIn ON movieTitle = title AND movieYear = year
RIGHT OUTER JOIN MovieStar ON MovieStar.name = StarsIn.starname
GROUP BY StarsIn.starname
HAVING COUNT(DISTINCT title) <= 5;
```
- г)

```
SELECT name, year(birthdate), COUNT(SELECT DISTINCT studioName
FROM Movie
JOIN StarsIn ON title = movieTitle AND year = movieYear
WHERE starname = name)
FROM MovieStar
HAVING COUNT(SELECT * FROM StarsIn WHERE starname = name) <= 5
ORDER BY name, year(birthdate);
```

Задача 5 (10т.). Текстов файл с име `comproc1` съдържа зададената по-долу последователност от команди на `bash` за Linux. Напишете вдясно какво ще бъде изведено на стандартния изход след стартиране на файла с команден ред

```
bash compoc1 3 4 5
```

ако на стандартния вход бъде подадена последователността от символи 6 7

```
br=0
br=`expr $br + $2`
set 3 5 7
shift
for j
do for var
    do if test $br -lt $2
        then br=`expr $br \* $2`
            echo $br $var $j >> file
        else br=`expr $br - $2`
            echo $var $br $j >> file
        fi
    done
    echo $*
    break
done
read a1 a2
until cat file | grep $a1
do echo $#
    grep $1 file
    exit
    echo END
done
echo $3
wc -l < file
echo FINAL
```


Задача 6 (10т.). Софтуерна система „Галерия” е предназначена за оферирание и продажба на произведения на изкуството като предоставя услуги на регистрирани и нерегистрирани потребители. Тя има следните изисквания:

1. Нерегистрирани и регистрирани потребители могат да разглеждат произведенията на изкуството, представени в системата.
2. Потребител би могъл да има само една регистрация. В нея се запазва информация за името на потребителя и уникален негов номер, както и се създава акаунт, който да съхранява информацията за предоставените и закупените произведения на автора.
3. Регистриран купувач през своята регистрация може да запазва за закупуване до 3 произведения на изкуството и с разрешение на автора на творбата.
4. Регистриран купувач през своята регистрация би могъл да закупи произведение на изкуството чрез интернет система за плащане.
5. Регистриран автор през своята регистрация може да предостави за продажба до 10 свои произведения. Той може да редактира информацията за тези произведения.
6. Творбите са разделени в три категории – картини, скулптури и графики.
7. Всяка творба е представена с име на творбата, автор, размер, цена и до 2 снимки. Предоставя се възможност за качване на информация, специфична за категорията на творбата.

А) Представете изискванията на системата чрез диаграма на случаите на употреба (Use Case Diagram) .

Б) Опишете изисквания 2, 5, 6, 7 чрез обектно-ориентиран модел на данните.

Задача 7 (10т.). Регулярен ли е езикът

$$L = \{ w \in \{0, 1\}^* \mid w \text{ съдържа точно веднъж } 010 \text{ като поддума} \}?$$

Обосновете отговора си!

Задача 8 (10т.). Пресметнете определения интеграл:

$$\int_0^{1/2} \arcsin x \, dx.$$

14.07.2015

СУ-ФМИ

Държавен изпит за ОКС
Бакалавър

**Софтуерно
инженерство**

ф.н. _____

лист 12/14

Ч Е Р Н О В А

14.07.2015

СУ-ФМИ

Държавен изпит за ОКС
Бакалавър

**Софтуерно
инженерство**

ф.н. _____

лист 13/14

Ч Е Р Н О В А

14.07.2015

СУ-ФМИ

Държавен изпит за ОКС
Бакалавър

**Софтуерно
инженерство**

ф.н. _____

лист 14/14

Ч Е Р Н О В А