

СОФИЙСКИ УНИВЕРСИТЕТ
„СВ. КЛИМЕНТ ОХРИДСКИ“ФАКУЛТЕТ ПО МАТЕМАТИКА
И ИНФОРМАТИКАДЪРЖАВЕН ИЗПИТ
ЗА ПОЛУЧАВАНЕ НА ОКС “БАКАЛАВЪР ПО ИНФОРМАЦИОННИ СИСТЕМИ”

ЧАСТ I (ПРАКТИЧЕСКИ ЗАДАЧИ)

12.07.2016 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 белава;
- Времето за работа по изпита е 3 часа.

Изпитната комисия ви пожелава успешна работа!

Задача 1. Дадена е квадратна матрица от дробни числа с размерност 11×11 . Казваме, че два елемента на матрицата са **симетрични**, ако те са разположени симетрично относно (1) главния диагонал, (2) вторичния диагонал, или (3) пресечната точка на двата диагонала на матрицата. На всеки елемент от матрицата съответстват най-много три симетрични елемента. Съвкупност, състояща се от елемент на матрицата заедно с всички негови симетрични елементи, наричаме **симетрична група**.

Пример: В матрицата с размерност 3×3

$$\begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{pmatrix}$$

$\{a_{01}, a_{10}, a_{21}, a_{12}\}$ и $\{a_{00}, a_{22}\}$ са симетрични групи.

Да се състави функция, която променя матрицата така, че **всеки** елемент a_{ij} , $0 \leq i, j \leq 10$ от матрицата се заменя със средно-аритметичната стойност на елементите в неговата симетрична група.

Задача 2. Задачата да се реши с използване на език за процедурно или обектно-ориентирано програмиране (C, C++ или Java).

Нека в текстов файл се съдържа единствен ред текст, в който е описано представяне на дърво по следната схема:

$\langle \text{дърво} \rangle ::= (\langle \text{корен} \rangle (\langle \text{празен низ} \rangle | \langle \text{списък от наследници} \rangle)),$

където

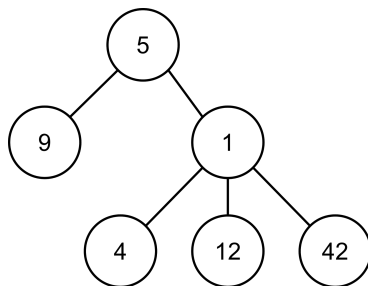
$\langle \text{корен} \rangle ::= \langle \text{цяло число} \rangle$

$\langle \text{списък от наследници} \rangle ::= \langle \text{дърво} \rangle | \langle \text{дърво} \rangle, \langle \text{списък от наследници} \rangle$

В текста се допуска наличието на произволен брой интервали между отделните елементи.

Да се напише функция, която получава като параметър низ, задаващ име на текстов файл, описващ дърво съгласно горната схема. Функцията да прочита съдържанието на файла, да построява в паметта дърво, съответстващо на текстовото описание, и да извежда на конзолата максималната сума от стойностите по път от корена на дървото до някое от листата му. Конкретното представяне на дървото в паметта е по Ваш избор.

Пример: Дървото, показано на долната диаграма, се представя по следния начин:



$(5 ((9 ()), (1 ((4 ()), (12 ()), (42 ())))))$

За този пример функцията следва да изведе на екрана числото $48 = 5 + 1 + 42$.

Задача 3. Нека са дадени следните изрази на езиците Haskell и Scheme. Да се посочи каква е оценката на изразите. Изберете само един от двата езика за решението на задачата и напишете името му в даденото за целта поле. Точки за задачата се дават само за изборения от Вас език.

Избран език:

Haskell:	
1.	<code>[x : [x] x <- [[1,2], [3,4]]]</code> Оценка:
2.	<code>[map (f 5) [1,2,3] f <- [(+), (-), (*)]]</code> Оценка:
3.	<code>"a" : [['b', 'c'], "d"]</code> Оценка:
Scheme:	
1.	<code>(map (lambda (x) (append (list x) x)) '((1 2) (3 4)))</code> Оценка:
2.	<code>(map (lambda (f) (map (lambda (x) (f 5 x)) '(1 2 3))) (list + - *))</code> Оценка:
3.	<code>(apply list (list (quote +) (quote 5) 8))</code> Оценка:

Задача 4. Дадена е базата от данни *Movies*, в която се съхранява информация за филми, филмови студиа, които ги произвеждат, продуцентите на филмите, както и актьорите, които участват в тях.

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

title – заглавие;

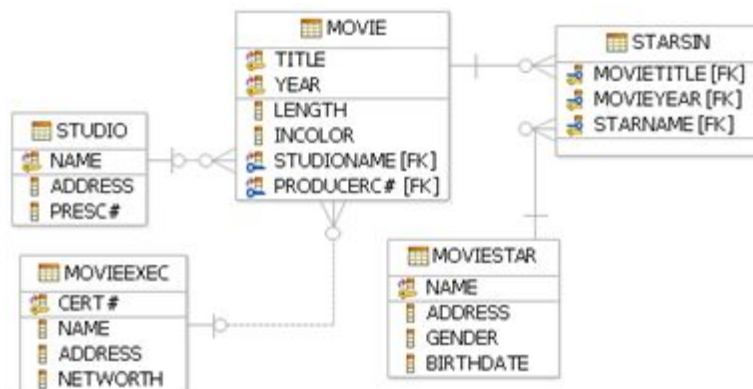
year – година, в която е заснет филмът;

length – дължина в минути;

incolor – 'Y' за цветен филм и 'N' за чернобял;

studioName – име на студио, външен ключ към таблицата *Studio*;

producerc# – име на продуцент, външен ключ към таблицата *MovieExec*.



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

name – име, първичен ключ;

address – адрес;

presc# – номер на сертификат на президента на студиото.

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

name – име, първичен ключ;

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 атрибута заедно формират първичния ключ. Атрибутите *movietitle* и *movieyear* образуват външен ключ към таблицата *Movie*.

movietitle – заглавие на филма;

movieyear – година на заснемане на филма;

starname – име на филмовата звезда, външен ключ към таблицата *MovieStar*.

Таблицата **MovieExec** съдържа информация за продуцентите на филми.

cert# – номер на сертификат, първичен ключ;

name – име;

address – адрес;

networth – нетни активи;

birthdate – рождена дата.

Забележка за всички таблици: Всички атрибути, които не участват във формирането на първичен ключ, могат да приемат стойност *null*.

1. Да се посочи заявката, която извежда името на продуцента и имената на филмите, продуцирани от продуцента на 'Pretty Woman'. Възможно е името на продуцента на филма да не е известно.

A)

```
SELECT t.name, title
FROM movie m JOIN (SELECT name, cert#
                    FROM movieexec
                    WHERE cert# IN (SELECT producerc#
                                    FROM movie
                                    WHERE title='Pretty Woman')) t
ON m.producerc#=t.cert#;
```

Б)

```
SELECT name, title
FROM (SELECT cert#
      FROM movieexec
      INTERSECT
      SELECT producerc#
      FROM movie
      WHERE title ='Pretty Woman') t
```

В)

```
SELECT name, title
FROM movie m JOIN movieexec me ON m.producerc#=me.cert#
WHERE title = 'Pretty Woman';
```

Г)

```
SELECT name, title
FROM movie m LEFT JOIN movieexec me ON m.producerc#=me.cert#
WHERE me.cert# NOT IN (SELECT producerc#
                      FROM movie
                      WHERE title='Pretty Woman');
```

Задача 5. Информационна система за градски транспорт, подобен на този в град София, съхранява информация за закупените от гражданите електронни карти за пътуване.

- Предлагат се два вида карти — карти за определен брой пътувания и карти за определени линии (например карта, която важи едновременно за метро №2, за трамвай №2 и за трамвай №9).
 - Картите за определени линии имат и период на валидност (начална и крайна дата), като крайната дата не е задължителна, т.е. картите могат да бъдат безсрочни.
 - Линиите на градския транспорт имат номер и тип (метро, трамвай, тролейбус).
 - Всяка карта има цена в левове (с точност до два знака след десетичната запетая).
 - Електронните карти се издават на пластични носители (така наричаме пластиката, която гражданите носят със себе си и показват на контрольорите).
 - Във всеки носител може да бъдат заредени няколко електронни карти, напр. в един и същ носител може да има карта за трамваи №1 и №5, валидна от 1.7.2016 г. до 31.7.2016 г., и карта за 10 пътувания.
 - Всеки носител има уникален идентификатор. Някои носители имат и име на собственика, а другите са анонимни.
1. Да се създаде и обоснове E/R модел на БД, която съхранява гореописаната информация. Да се начертае E/R диаграма на модела.
 2. Да се преобразува E/R диаграмата към релационни схеми. Да се премахнат излишествата, където това е възможно.
 3. Да се напише DDL код, съответстващ на релационните схеми. Да се реализират всички описани ограничения. За тип на линия да не се допускат други стойности, освен изброените (метро, трамвай и тролейбус).

Задача 6. Разглеждаме информационна система за карти за градския транспорт. За нея знаем че:

- Картите са за различни периоди — един, три, шест месеца, както и от различен тип: за една линия, две линии или за цялата градска мрежа.
- Всяка конкретна карта се характеризира с номера си и не е поименна.
- Поддържа се електронна онлайн услуга, чрез която може да се провери състоянието (валидна/невалидна) на всяка карта или да се зареди за определен период. В тази система е възможно да се промени типът на карта (например от една линия към всички линии с доплащане) или да се разблокира.
- Системата поддържа и услуги за контролорите чрез достъп от специално устройство. Те могат да проверят състоянието на карта или да я блокират при неправомерно използване.
- Разблокирането става след заплащане на определената глоба.

1. Да се направи пълно описание на потребителски случай (use case), който представя как потребител зарежда карта за градския транспорт.

В описанието да се включат минимум: актьори, които участват; предусловия; постусловия; основен сценарий; алтернативни сценарии; взаимодействие с други потребителски случаи (ако е приложимо, но само с препратка); бизнес обекти, с които се работи, и евентуални промени в състоянието им; бизнес правила и/или условия, които се проверяват при обработката.

2. Да се опишат и нефункционалните изисквания към потребителския случай, свързани с производителност (performance) и използване (usability).

Задача 7. Разглеждаме отново описанието на информационната система от задача 6 (система за карти за градския транспорт).

Да се нарисува диаграма на състоянието на картата (State Machine Diagram). Към диаграмата да се добави кратко текстово описание.

Задача 8. Да се пресметне определеният интеграл

$$\int_1^2 x \log_2 \frac{1}{x} dx$$

12.07.2016

СУ-ФМИ

Държавен изпит за ОКС
Бакалавър

Информационни
системи

ф.н. _____

лист 12/12

ЧЕРНОВА