

**СОФИЙСКИ УНИВЕРСИТЕТ  
„СВ. КЛИМЕНТ ОХРИДСКИ“****ФАКУЛТЕТ ПО МАТЕМАТИКА  
И ИНФОРМАТИКА****ДЪРЖАВЕН ИЗПИТ  
ЗА ПОЛУЧАВАНЕ НА ОКС “БАКАЛАВЪР ПО ИНФОРМАЦИОННИ СИСТЕМИ”****ЧАСТ I (ПРАКТИЧЕСКИ ЗАДАЧИ)  
14.7.2015 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.** (10 т.) *Задачата да се реши на езика C++ или Java. В началото на вашето решение посочете кой език сте избрали.*

Даден е двумерен масив `img`, с елементи от тип `float`, с размери  $M (\leq 10)$  реда и  $N (\leq 10)$  стълба.

Напишете функция `subsample`, която получава като аргументи  $M$ ,  $N$  и масива `img` и която създава и извежда на екрана нов двумерен масив  $S$ , за който е изпълнено:

1.  $S$  има размери  $(M+1)/2$  и  $(N+1)/2$
2. Всеки елемент  $S[i][j]$  е равен на средно-аритметичното от елементите на входния масив разположени на позиции  $M[y][x]$ , където  $i*2 \leq y \leq i*2+1$  и  $j*2 \leq x \leq j*2+1$ .

Например при масив `img`, представен таблично така

```
1.0 2.0 3.0
4.5 6.5 7.5
```

функцията би извела 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.** (10 т.) *Задачата да се реши на езика C++ или Java. В началото на вашето решение посочете кой език сте избрали.*

*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)

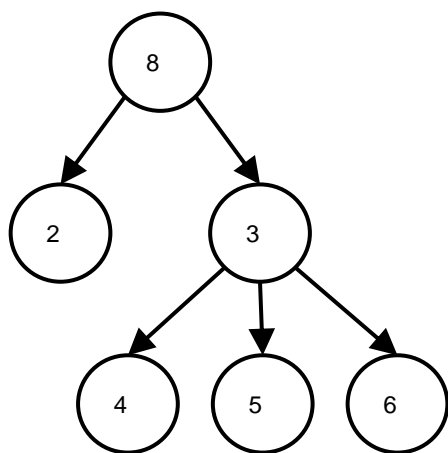
```
bool member ([подходящ тип]root, int x)
```

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

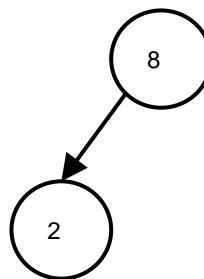
в) Да се дефинира рекурсивна функция (или статичен метод)

```
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.** (10 т.) При решението на задачата попълнете или само частта за Scheme или само частта за Haskell. По-долу на листа е предвидено място, в което да посочите кой език сте избрали. Ще се оценяват само и единствено решенията за езика, който сте посочили в това поле. Ако не посочите език, задачата се оценява с нула точки.

Нека са дадени следните изрази на езиците Haskell и Scheme. Моля, посочете каква ще бъде тяхната оценка за един от двата езика (по ваш избор).

Избран от вас език:

### Haskell

```
map (head [(\\couple->fst couple + snd couple)])  
      (foldr1 (++) [[(1,2)],[(3,4)]])
```

Оценка: \_\_\_\_\_

```
[zip [x] [x] | x <- [1..5]]
```

Оценка: \_\_\_\_\_

```
map (\\(x:y:z)->x:z) [[1,2,3],[2,3,1],[3,1,2]]
```

Оценка: \_\_\_\_\_

### Scheme

```
(map  
  (car (list (lambda (couple) (+ (car couple) (cdr couple)))))  
  (apply append '( ( (1 . 2) ) ( (3 . 4) ) ) ) )
```

Оценка: \_\_\_\_\_

```
(map (lambda (x)  
      (cons x (list x)))  
      '(1 2 3 4 5))
```

Оценка: \_\_\_\_\_

```
(map (lambda (pred) (filter pred '(1 2 3 4 5)))  
      (list even? odd?))
```

Оценка: \_\_\_\_\_

**Задача 4.** (10 т.) Информационна система съхранява информация за обяви за продажби на недвижими имоти и за агенциите, които ги предлагат.

Поддържат се два вида обяви - за апартамент и за парцел. За всяка обява задължително се съхранява следната информация: уникален идентификатор, описание (до 1000 символа), град (до 30 символа), квартал (до 30 символа), цена (с точност до два знака след десетичната запетая), площ в кв. м. (цяло число). За някои обяви е посочен и допълнителен ориентир (до 50 символа, напр. “бул. Джеймс Баучър”, “НДК” и т.н.).

Във всяка обява за апартамент допълнително са указани номер на етаж и брой стаи, а във всяка обява за парцел - дали е в регулация.

Всяка обява се предлага от агенция. За всяка агенция задължително се съхранява уникален идентификатор, име (до 50 символа) и уеб сайт (низ, започващ с “http://” или “https://”). Една агенция може да предлага много обяви. Две агенции не могат да имат еднакъв сайт.

- а) Създайте E/R модел на БД, която съхранява гореописаната информация.
- б) Преобразувайте E/R диаграмата към релационни схеми. Премахнете излишествата, където това е възможно.
- в) Напишете DDL код, съответстващ на релационните схеми. Реализирайте всички описани ограничения.

**Задача 5. (10 т.)** 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.

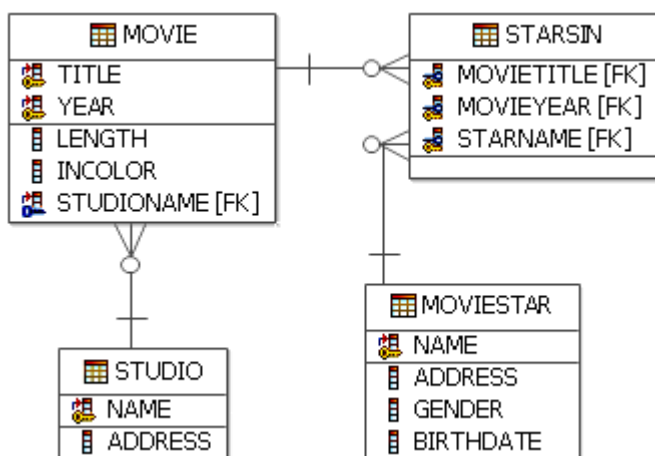
*title* – заглавие;

year – година, в която филмът е заснет;

*length* – дължина в минути;

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

*studio*name – име на студио, външен ключ.



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

*name* – имя;

*address* – адрес;

*gender* – пол, 'М' 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 изведат имената и рождените дати на всички филмови звезди, чието име не завършва на “а” и са играли както в цветни, така и в черно-бели филми. Първо да се изведат най-младите звезди, а звезди, родени на една и съща дата, да се изведат по азбучен ред.

**2.** За всяка филмова звезда, играла в най-много 5 филма, да се изведе следната информация: име; рождена година (напр. ако звездата е родена на 1.1.1980 г., в колоната да пише 1980); брой студия, с които е работила.

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

**Задача 6.** (10 т.) Представете си система за онлайн анкети, подобна на Doodle. Целта на системата е да подпомага вземането на решения, например, за определяне на дата за изпит сред студенти или дали да се организира тийм-билдинг в една фирмата или да се дадат допълнителни финансови бонуси. Изисква се в една анкета да могат да участват или всички членове на дадена организация (фирма, университет), или определено тяхно подмножество (отдел или група), или външни хора, които ще получат на електронната си поща писмо, с еднократна връзка към форма за попълване на анкетата.

Представете модел на потребителските случаи за тази система и го опишете текстово. Включете поне 7 потребителски случая с евентуални връзки между тях. Дефинирайте актьорите и обяснете техните роли, посочете границите на системата, евентуална връзка с външни системи.

**Задача 7.** (10 т.) Разгледайте отново условието на задача 6. След това решете следните задачи:

- a) (5 т.) Представете UML диаграма на състоянието (state-machine) за една анкета. Включете поне 4 различни състояния. Опишете преходите към тях.
- b) (5 т.) Направете пълно описание на потребителски случай (use case), който описва попълване на анкета. Потребителският случай трябва да описва и какви проверки за коректност се правят при обработката (напр. дали лицето има права и др.). Опишете и нефункционалните изисквания към потребителския случай.



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

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

14.7.2015 г.

СУ-ФМИ

Държавен изпит  
за ОКС *Бакалавър*

**Информационни  
системи**

ф.н.

лист  
10/12

ЧЕРНОВА

14.7.2015 г.

СУ-ФМИ

Държавен изпит  
за ОКС *Бакалавър*

**Информационни  
системи**

ф.н.

лист  
11/12

ЧЕРНОВА

14.7.2015 г.

СУ-ФМИ

Държавен изпит  
за ОКС *Бакалавър*

**Информационни  
системи**

ф.н.

лист  
12/12

ЧЕРНОВА