

СОФИЙСКИ УНИВЕРСИТЕТ  
“СВ. КЛИМЕНТ ОХРИДСКИ”



ФАКУЛТЕТ ПО МАТЕМАТИКА И  
ИНФОРМАТИКА

**Д Ъ Р Ж А В Е Н И З П И Т**  
ЗА ПОЛУЧАВАНЕ НА ОКС “БАКАЛАВЪР” ПО Софтуерно инженерство

ЧАСТ I (ПРАКТИЧЕСКИ ЗАДАЧИ)

11. 07. 2017 г.

Време за работа – 3 часа

*Драги абсолвенти, спазвайте стриктно следните указания:*

- ☐ Пишете само на предоставените ви листове без да ги разкопчавате
- ☐ Попълнете горе вдясно **ФАКУЛТЕТНИЯ СИ НОМЕР В ПОЛЕТО НА ВСЯКА НЕЧЕТНА СТРАНИЦА**
- ☐ Решението на всяка задача се разполага в предвиденият за това лист
- ☐ При необходимост пренасяте решението на подпечатан нов лист, предоставян от квесторите
- ☐ Не се допуска използването на персонални електронни устройства.

*И з п и т н а т а   к о м и с и я   в и   п о ж е л а в а   у с п е ш н а   р а б о т а .*

**Задача 1. (30 min)**

Горски терен е представен с помощта на мрежа или двумерен масив с  $m \times n$  ( $m, n \in [0; 100]$ ) области (елементи). Елементите на двумерния масив са символи, които имат следното значение:

- **R** – река;
- **S** – скала;
- цифри от **1** до **4**, които означават гъстота на гората.

Теренът се променя на всеки 10 години. Реката и скалите остават постоянни, докато гъстотата на горските площи се променя по следните правила:

- **1** преминава в **2**, **2** в **3**, а **3** в **4**.
- **4** преминава в **3**, ако в съседство има поне 3 области с гъстота **4**, в противен случай остава **4**.

Съседни на дадена област (елемент) от масива са тези области, индексите на които се различават най-много с 1 т.е. всяка област има най-много 8 съседни области.

Напишете функция, която по подаден терен намира вида му след 100 години.  
Демонстрирайте използването на функцията в кратка програма.

Примерен вид на терен:

```
R R 1 1 2 2
1 R R R 1 2
S 1 R R 2 3
4 4 S S R R
```



**Задача 2. (25 min)**

Напишете функция, която получава като аргумент свързан списък с елементи цели числа и го сортира. Списъкът е представен и подаден на функцията по избран от вас начин.

Напишете кратка програма, която създава списък, добавя в него няколко елемента, използва функцията за сортиране и извежда на екрана резултата.

Използвайте езиките JAVA, C или C++, но без библиотечни структури данни и алгоритми.



**Задача 3. (25 min)**

Дадена е базата от данни **Movies**, в която се съхранява информация за филми, филмови студии, 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

*Studio.name*;

*producerc#* - номер на сертификат на продуцента, външен ключ към *MovieExec.cert#*.

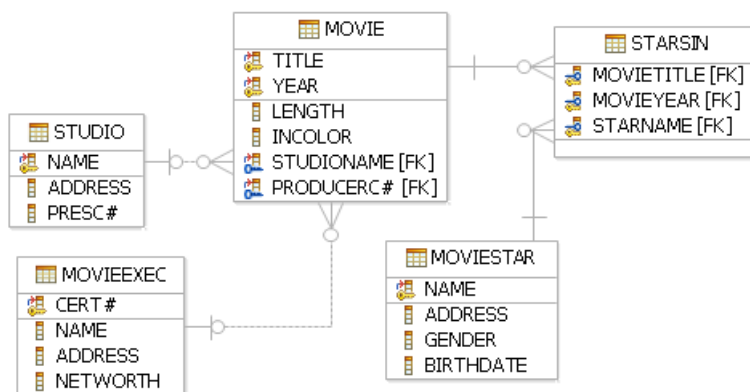
*cert#*.

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

*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 *Movie*.

*movietitle* – заглавие на филма;

*movieyear* – година на заснемане на филма;

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

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

*cert#* – номер на сертификат, първичен ключ;

*name* – име;

*address* – адрес;

*networth* – нетни активи;

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

1. Попълнете липсващите части, обозначени с \_\_\_\_\_ така, че заявката да изведе за всяко студио името на студиото, заглавието и годината на филма, излязъл последно на екран за това студио.

```

SELECT studioName, title, year
FROM movie m
WHERE year = (SELECT _____
               FROM movie
               WHERE _____);
  
```

2. Попълнете липсващите части, обозначени с \_\_\_\_\_ така, че заявката да изведе име на продуцент и обща дължина на продуцираните от него филми, за тези продуценти, които имат поне един филм преди 1980 г.

```
SELECT name, _____
FROM movieexec JOIN movie ON producerc# = cert#
_____
_____;
```

3. Попълнете липсващите части, обозначени с \_\_\_\_\_ така, че заявката да изведе име на актьорите, участвали във филми на продуценти с най-големи нетни активи, както и заглавие на филмите, в които са участвали, име на продуцент и нетни активи.

```
SELECT starname, title, name, networth
FROM starsin JOIN movie ON movietitle=title AND movieyear=year
      JOIN      (SELECT cert#, networth, name
                  FROM movieexec
                  WHERE _____) t
_____;
```

4. Заградете буквата на заявката, която извежда името на продуцента, заглавието и годината на всички филми, продуцирани от продуцента на филма 'Interstellar'.

A)

```
SELECT name, title, year
FROM movie, movieexec
WHERE producerc#=cert# AND title='Interstellar' AND cert# IN (SELECT producerc#
                                                                FROM movie
                                                                WHERE title='Interstellar');
```

Б)

```
SELECT t.name, title, year
FROM movie m JOIN (SELECT name, cert#
                    FROM movieexec
                    WHERE EXISTS (SELECT producerc#
                                  FROM movie
                                  WHERE title='Interstellar')) t
ON m.producerc#=t.cert#;
```

В)

```
SELECT name, title, year
FROM movie JOIN movieexec ON producerc#=cert#
WHERE cert# = ANY (SELECT producerc#
                   FROM movie
                   WHERE title='Interstellar');
```

Г)

```
SELECT DISTINCT name, movietitle, movieyear
FROM movie JOIN movieexec ON producerc#=cert#
      JOIN starsin ON year=movieyear AND title=movietitle
WHERE cert# IN (SELECT producerc#
                FROM movie
                WHERE title='Interstellar');
```

**Задача 4. (25 мин.)**

В текущия каталог се намира текстов файл file.txt със следното съдържание

```
abcdef
0123456789
ABCD
```

Изпълнимият файл, получен след компилация на зададения по-долу програмен фрагмент, се стартира с командния ред:

```
./a.out ff
```

Напишете какво ще бъде изведено на стандартния изход и какво ще бъде съдържанието на двата файла след приключване на успешното изпълнение

```
#include <stdio.h>
#include <fcntl.h>
main( int argc, char * argv[])
{
    int fdi, fdo, k, broi, i = 0 , status;
    char buff [ 40 ], c ;
    if ( ( fdi = open ("file.txt", O_RDWR ) ) == -1 )
        { printf ("\n Cannot open \n" ); exit (1); }

    if ( ( fdo= open (argv[1], O_CREAT| O_TRUNC| O_RDWR, 0666) ) == -1 )
        { printf ("\n Cannot open \n" ); exit (1); }

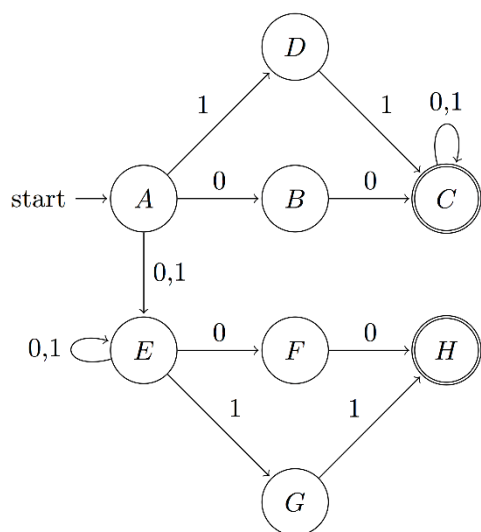
    if ( fork( ) == 0 )
    {
        k=dup(1); close(1); dup(fdi);
        broi = read ( fdi , buff, 40 );
        c = buff[ i++];
        if ( c <= '0' || c >= '9' )
        {
            while ( buff [ i ++] != '\n' && i < broi )
                write ( 1, "*", 1 );
            write ( 1, "\n", 1 );
            close(1); dup(k);
            write ( 1, buff, 3 );
            write ( fdo, buff, 10 );
        }
        else { write( 1, buff, broi ); close(1);dup(k);
              write ( 1, "*\n", 2 ); }
        lseek( fdo,0, 0);
        write ( fdo, "*\n", 2 );
        close ( fdi); close (fdo);
    }
    else { wait ( &status);
          close(1); dup(fdi);
          execlp ("grep", "grep", "c", argv[1], 0 );
          execlp ("wc", "wc", "-l", "ff", 0 );
        }
}
```





**Задача 5. (20 мин.)**

Да се намери минималният краен детерминиран автомат, еквивалентен на автомата





**Задача 6. (20 мин.)**

Цялостният софтуерен продукт е on-line платформа, която свързва енергийни дружества-дистрибутори на електроенергия и клиенти-потребители на електроенергия. Формулирани са следните изисквания на системата:

1. Системата изисква задължителна регистрация на всички потребители в съответните им роли.
2. Системата ще позволява регистриран дистрибутор на енергия да обработва информация за своите клиенти, а именно запис на данни за клиента, запис на количеството употребена енергия за текущия отчетен период, както и съхранение на клиентска история.
3. Системата ще позволява регистриран дистрибутор да оферира енергия и да издава фактури за заплатена от потребителя енергия.
4. Системата ще дава възможност на регистриран потребител на енергия да преглежда офертите на дистрибуторите. Този потребител ще може да се информира за текущо използваната на енергия като допълнително ще има възможност да се запознае с историята на своето потребление.
5. Системата ще предоставя възможност на регистриран потребител на енергия да заплаща енергия чрез външна система за електронно разплащане.

Съставете диаграма на потребителските случаи на описаната система.



**Задача 7. (25 мин.)**

Софтуерна система за калкулиране на отстъпки предоставя следната функционалност:

1. Нов клиент, който получава нова кредитна карта получава 15 % отстъпка от покупките си.
2. Съществуващ клиент, който е притежател на карта за лоялен клиент, получава 10% отстъпка.
3. Клиент, притежаваш ваучер, получава 20% отстъпка, като не може да се възползва от отстъпката за нов клиент.

Дефинирайте таблица за взимане на решения, въз основа на която могат да се генерират тестови сценарии. Таблицата трябва да показва условията, следствията от тях и правилата, от които се генерират тестовите сценарии.



Ч Е Р Н О В А