

СОФИЙСКИ УНИВЕРСИТЕТ
“СВ. КЛИМЕНТ ОХРИДСКИ”



ФАКУЛТЕТ ПО МАТЕМАТИКА
И ИНФОРМАТИКА

ДЪРЖАВЕН ИЗПИТ

ЗА ПОЛУЧАВАНЕ НА ОКС “БАКАЛАВЪР ПО ИНФОРМАЦИОННИ СИСТЕМИ”

ЧАСТ I (ПРАКТИЧЕСКИ ЗАДАЧИ)

Драги абсолвенти:

- Попълнете факултетния си номер в горния десен ъгъл на всички листове.
- Пишете само на предоставените листове, без да ги разкопчавате.
- 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 имате нужда от допълнителен лист, можете да поискате от квесторите.
- На един лист не може да има едновременно и чернова, и белова.
- Черновите трябва да се маркират, като най-отгоре на листа напишете “ЧЕРНОВА”.
- Ако решението на една задача не се побира на нейния лист, трябва да поискате нов бял лист от квесторите. Той трябва да се защити с телбод към листа със задачата.
- Всеки от допълнителните листове (белова или чернова) трябва да се надпише най-отгоре с вашия факултетен номер.
- Черновите също се предават и се защитават в края на работата.
- Времето за работа по изпита е 3 часа.
- Всяка напълно и коректно решена задача, съгласно критериите за оценяване, носи 10 точки.
- Оценката от практическата част се изчислява по формулата $\frac{x}{10}$, където x е сумата от получените точки от задачите. Максималната оценка е Отличен 6.00.
- За успешно полагане на изпита са необходими поне 30 точки, като при събрани под 30 точки оценката е Слаб 2.00.

Изпитната комисия ви пожелава успешна работа!

Задача 1. Решете задачата на езика C++. Решения на друг език носят нула точки. Отговорите на подточки 1А, 1Б, 1В и 1Г трябва да съвпадат с това, което би извела програмата. При несъответствие отговорът се оценява с нула точки. Решението на подточка 1Д трябва да бъде технически издържано (например не бива да изтича памет, трябва да се спазват добрите практики за структуриране на програмата и т.н.). Ако решението съдържа сериозни грешки, то се оценява с нула точки.

1А) (1 точка) Какъв ще бъде изходът от изпълнението на следния фрагмент:

```
int calc(int a[5])
{
    int sum = 8;
    for (int i = 0; a[i]; ++i)
        sum += i;
    return sum;
}
```

```
int a[7] = {1, 2, 3};
std::cout << calc(a);
```

Отговор: _____

1Б) (1 точка) Какъв ще бъде изходът от изпълнението на следния програмен фрагмент?

```
char text[] = "hello", *p = text;
while (*p) std::cout << ++*p++;
```

Изберете един от следните отговори:

- а) Грешка по време на компилация.
- б) Грешка по време на изпълнение.
- в) ifmmp
- г) el
- д) Безкраен цикъл.

1В) (1 точка) Какво ще изведе следният фрагмент (приемаме, че е част от валидна програма):

```
int *pt;
int a[3] = {4, 19, 13};
pt = &a[1];
pt += 1;
std::cout << *pt << std::endl;
```

Отговор: _____

1Г) (1 точка) В дадените по-долу празни места попълнете какви ще бъдат стойностите на елементите на двата масива А и В след обръщението към функцията f.

```
void f(int * arr1, const int * arr2)
{
    int *p1 = arr1;
    const int *p2 = arr2;

    while(*p2 >= 0)
    {
        *p1++ = *p2++;
    }
}
```

```
void main()
{
    int A[4] = {-1, -2, -3, -4};
    int B[4] = {10, 20, 30, -1};

    f(A, B);
}
```

Отговор:

A[0] = ____ A[1] = ____ A[2] = ____ A[3] = ____

B[0] = ____ B[1] = ____ B[2] = ____ B[3] = ____

1Д) (6 точки) Дадени са структура `Point`, описваща точка в декартова координатна система с координати `x` и `y` от тип `float`, и структура `Circle`, описваща окръжност с център `center` от тип `Point` и радиус `r` от тип `float`.

Да се дефинира функция `findRelativePosition`, която определя относителната позиция на две дадени окръжности една спрямо друга. Резултатът от изпълнението на функцията е стойност от изброения тип:

```
RelativePosition {NO_COMMON_POINTS, TOUCHING,  
                 INTERSECTING, SAME}.
```

със следния смисъл:

- `NO_COMMON_POINTS`: без общи точки
 - `TOUCHING`: допиращи се
 - `INTERSECTING`: пресичащи се
 - `SAME`: съвпадат
-

Задача 2. Задачата да се реши на езика C++. Отговорите си попълнете в посочените за тях полета. Оценява се само попълненото в полетата. Всичко друго писано по листа не носи точки.

2А) (6 точки) Разгледайте програмата. Някои от конструкциите във функцията main са номерирани в коментар. За всеки такъв ред посочете какво ще се изведе в резултат от неговото изпълнение. Отговор, който не съответства точно на това, което извежда програмата, носи нула точки. Ако смятате, че някой от редовете няма да изведе нищо, напишете "нищо". Ако смятате, че някой от тях ще предизвика грешка, опишете каква. Ако за някоя конструкция не бъде попълнено нищо, отговорът не носи точки.

```
#include <iostream>
using std::cout;

class test {
public:
    int var;

    test() { cout << "test()\n"; var = 0; }

    test &operator=(const test &other) {
        if (this != &other) {
            cout << "copy\n";
            var = other.var;
        } else {
            cout << "self-assignment\n";
        }
        return *this;
    }

    static test &instance() {
        static test obj;
        return obj;
    }

    test &self() { return *this; }
    void _oncreate() { cout << "_oncreate()\n"; }
    void _oncopy() { cout << "_oncopy()\n"; }
};

int main() {
    test &r1 = test::instance(); // 1
    test &r2 = test::instance(); // 2
    r1.var = 10; //3
    cout << r2.var << "\n"; // 4
    r1.self() = r2; // 5
    new test[3]; // 6
}
```

Посочете отговорите си тук:

(1 точка) Ред //1 ще изведе:

(1 точка) Ред //2 ще изведе

(1 точка) Ред //3 ще изведе:

(1 точка) Ред //4 ще изведе

(1 точка) Ред //5 ще изведе

(1 точка) Ред //6 ще изведе

2Б) (1 точка) Възможно ли е в един клас да се дефинират няколко различни копиращи конструктора?

а) Да

б) Не

2В) (3 точки) Нека са дадени следните дефиниции:

```
class foo {
public:
    foo(int) {}
};

void g(foo) {}
void f(int) {}
```

Срещу всеки от изразите посочете (като напишете "да"или "не") дали ще се компилира:

g(5); _____

f(foo(5)); _____

foo('a'); _____

Задача 3. Задачата да се реши на езика *Haskell*. Отговорите си попълнете в посочените за тях полета. Оценява се само попълненото в полетата. Всичко друго писано по листа не носи точки.

Под всеки от дадените по-долу изрази посочете каква ще бъде неговата оценка. За някои от изразите са дадени възможности, измежду които трябва да изберете. Други трябва да съобразите и да попълните сами. За изразите, които трябва да попълните, точки се дават само за напълно коректни отговори. Ако посоченият от вас отговор не съответства точно на това, което би извел *Haskell* интерпретаторът, той се оценява с нула точки.

А) (1 точка) Оградете оценката на израза:

`foldr1 (&&) [False, False ..]`

- а) True
- б) False
- в) Изразът ще предизвика безкрайно изпълнение.

Б) (1 точка) Оградете оценката на израза:

`foldr1 (&&) [True, True ..]`

- а) True
- б) False
- в) Изразът ще предизвика безкрайно изпълнение.

В) (1 точка) Попълнете оценката на израза:

`filter (`elem` [10..20]) [1,5,10,100,20,15]`

Отговор: _____

Г) (1 точка) Оградете оценката на израза:

`negate $ max 10 20`

- а) 10
- б) 20
- в) -10
- г) -20
- д) Операторът \$
- е) Операторът \$ не може да се използва инфиксно.

Д) (1 точка) Попълнете оценката на израза:

`print (take 4 [1,3..])`

Отговор: _____

Е) (1 точка) Попълнете оценката на израза:

`(:[]) []`

Отговор: _____

Ж) (1 точка) Попълнете оценката на израза:

`map ($ 0) (map (+) [1..5])`

Отговор: _____

З) (1 точка) Нека имаме следната функция:

`f l = [x+y | x <- l, y <- l]`

Попълнете каква ще бъде оценката на израза:

`f [1,10]`

Отговор: _____

И) (1 точка) Нека имаме следната функция:

`g ([]:_) = []`

`g x = (map head x) : g (map tail x)`

Попълнете каква ще бъде оценката на израза:

`g [[1,2,3],[4,5,6]]`

Отговор: _____

Й) (1 точка) В полето по-долу попълнете какъв е типът на функцията `g` от предишния въпрос. Опишете го така, че функцията да може да работи само и единствено за списъци от списъци, съдържащи елементи от класа `Num`.

Отговор: `g ::` _____

Задача 4. Дадена е базата от данни **Movies**.

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

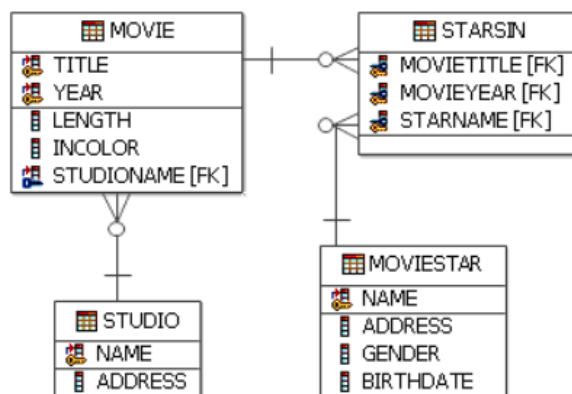
- *name* — име, първичен ключ;
- *address* — адрес.

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, първичен ключ;
- *address* — адрес;
- *gender* — пол, 'F' за жена и 'M' за мъж;
- *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.

Задача 4.1 (4 точки): Посочете заявката, която извежда имената на всички студиа, които имат поне един цветен филм и едновременно с това поне един филм с неизвестна дължина:

а) `SELECT studioName
FROM Movie
WHERE inColor != 'Y'
AND studioName =
(SELECT studioName
FROM Movie
WHERE length IS NULL);`

в) `SELECT studioName
FROM Movie
WHERE inColor = 'Y'
INTERSECT
SELECT studioName
FROM Movie
WHERE length IS NULL;`

б) `SELECT studioName
FROM Movie
WHERE inColor = 'Y'
AND length = NULL;`

г) `SELECT name
FROM Studio
LEFT JOIN Movie
ON name = studioName
WHERE inColor = 'Y'
OR length = NULL;`

Задача 4.2 (6 точки): Посочете заявката, която за всяка актриса извежда името и броя на черно-белите филми, в които е участвала. Ако за дадена актриса няма информация в какви филми е участвала или е играла само в цветни филми, срещу нейното име да се изведе числото 0.

a) SELECT name, COUNT(DISTINCT title)
FROM Movie
RIGHT JOIN StarsIn
ON title = movieTitle
AND year = movieYear
RIGHT JOIN MovieStar
ON starName = name
WHERE gender = 'F'
GROUP BY name
HAVING inColor = 'N';

б) SELECT name, COUNT(title)
FROM MovieStar
LEFT JOIN StarsIn ON name = starName
LEFT JOIN Movie
ON movieTitle = title
AND movieYear = year
AND inColor = 'N'
WHERE gender = 'F'
GROUP BY name;

в) SELECT name, COUNT(title)
FROM MovieStar
LEFT JOIN StarsIn ON name = starName
LEFT JOIN Movie
ON movieTitle = title
AND movieYear = year
HAVING inColor = 'N'
AND gender = 'F'
GROUP BY name;

г) SELECT starName, COUNT(*)
FROM StarsIn
LEFT JOIN Movie
ON title = movieTitle
AND year = movieYear
WHERE inColor = 'N' AND gender = 'F'
GROUP BY name;

Задача 5. Информационна система съхранява информация за ваксинации извън имунизационния календар. В системата се палят данните на гражданите, които искат да се ваксинират:

- ЕГН (или ЛНЧ) — низ от точно 10 символа, уникален за всеки гражданин
- име — низ до 50 символа
- e-mail — низ до 20 символа
- телефон — низ от точно 12 символа

В системата се съхраняват и данни за ваксините:

- име на ваксина — низ до 20 символа, уникално за всяка ваксина
- брой дози, които трябва да се поставят от тази ваксина — цяло положително число
- описание на ваксината — низ до 100 символа

Гражданинът прави резервация за поставяне на ваксина в център за ваксиниране. За центъра се пази следната информация:

- код на центъра — низ от точно 5 символа, уникален за всеки център
- име на центъра — низ до 20 символа
- адрес на центъра — низ до 50 символа

Центровете за ваксиниране могат да бъдат центрове в сграда (болница, университет, РЗИ и т.н.) или полеви (на открито, стадион, парк и т.н.).

Сградните центрове имат допълнителни характеристики: номер на етаж и номер на стая — цели положителни числа.

Полевите центрове имат допълнителна характеристика: локация (например: “Южен парк”) – низ до 20 символа.

При резервацията, гражданинът указва:

- дата, когато желае да бъде ваксиниран
- ваксина, с която желае да бъде ваксиниран
- център, в който желае да бъде ваксиниран

При резервацията в базата от данни трябва да се пази и информация за състояние на резервацията — цяло положително число (1 — действителна, 0 - отменена).

В центровете за ваксиниране работят лекари, които поставят ваксини. Лекарите се характеризират с:

- номер на лекар — низ точно 5 символа, уникален за всеки лекар
- име на лекар — низ до 50 символа

В центровете могат да работят много лекари, но един лекар работи само в един център.

В базата от данни трябва да се съхранява информация и ако на гражданина му е поставена ваксина.

Пази се информация за:

- дата, на която на гражданина му е поставена ваксината
- дата на която трябва да му бъде поставена следващата доза от ваксината (ако има такава, иначе NULL)
- кой лекар е поставил ваксината
- в кой център е поставена ваксината
- състояние на ваксинацията – цяло положително число (0 – не е завършена, 1 – завършена)

Задание:

- (4 точки) Да се направи E/R модел на БД, която съхранява гореописаната информация. Начертайте E/R диаграма на модела
- (3 точки) Преобразувайте E/R диаграмата към релационни схеми. Премахнете излишества, където това е възможно
- (3 точки) Напишете DDL код, съответстващ на релационните схеми. Реализирайте всички описани ограничения

Задача 6. В контекста на информационната система за ваксинации, представена в Задача 5 (за E/R моделиране), разглеждаме процес за регистриране на извършена имунизация от лекар:

- Регистрират се планови и извънпланови имунизации с данните, посочени в Задача 5 (за E/R моделиране).
- При успешна регистрация се генерира национален уникален номер на имунизацията и данните се записват в Електронното медицинско досие на пациента, част от Националната здравно-информационна система.
- След пълен ваксинационен цикъл на пациента може да се издаде сертификат за ваксинация.
- В случай, че поставената доза на ваксината предоставя на пациента сертификат за ваксинация, сертификатът се генерира от системата и може да бъде разпечатан.

Задание:

- а) (6 точки) Опишете в пълен формат потребителски случай за регистриране на имунизация. В описанието да се включат минимум: актьори; предусловия; следусловия (резултати); основен сценарий; отношения с други потребителски случаи (ако е приложимо); алтернативен сценарий; какви проверки за коректност се правят при обработката; специфични за потребителския случай допълнителни изисквания.
- б) (4 точки) В контекста на описанието на потребителския случай, изготвено в подусловие а), да се подготви диаграма на дейността (activity diagram) в термините на UML стандарта.

Ако при изпълнението на някое подусловие са направени допускания във връзка с представения сценарий, те трябва да бъдат описани явно.

Задача 7. (10 точки) Намерете неопределения интеграл

$$\int \operatorname{arctg} \sqrt{x} dx, \quad x > 0.$$

Чернова