

СОФИЙСКИ УНИВЕРСИТЕТ "СВ. КЛИМЕНТ ОХРИДСКИ"
ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА

ДЪРЖАВЕН ИЗПИТ
ЗА ПОЛУЧАВАНЕ НА ОКС "БАКАЛАВЪР ПО ИНФОРМАТИКА"
20-21.09.2003 г.

ЧАСТ I (ПРАКТИЧЕСКИ ЗАДАЧИ)

Задача 1

1) **{2 т.}** Какъв е броят на булевите функции на n променливи, принадлежащи на всяко от множествата:

A. $(T_0 \setminus T_1) \cap L$

- a) $n \cdot 2^n$ b) 2^{n-1} c) $2^{2^{n-1}}$ d) $3 \cdot 2^{n+1}$

B. $M \cap L \setminus (T_0 \cap T_1)$

- a) 0 b) n c) 2 d) $2^n - 2$

2) **{5 т.}** Дадено е следното множество от булеви функции:

$$\{ f_1(\tilde{x}^3) = (x_1 x_2 + x_1 \bar{x}_3 + x_2 \bar{x}_3) \rightarrow 0; f_2(\tilde{x}^3) = x_1 \downarrow ((x_1 \rightarrow \bar{x}_2) \vee \bar{x}_3);$$

$$f_3(\tilde{x}^3) = x_1 x_2 \vee x_1 \bar{x}_2 \bar{x}_3; f_4(\tilde{x}^3) = (x_1 \vee x_2) x_2 + x_3(x_3 + 1); f_5(\tilde{x}^3) = \overline{(x_1 \vee x_2)} \wedge \bar{x}_3 \}$$

A. Попълнете в таблицата вектор-стълба на всяка от горните функции:

x_1	x_2	x_3	$f_1(\tilde{x}^3)$	$f_2(\tilde{x}^3)$	$f_3(\tilde{x}^3)$	$f_4(\tilde{x}^3)$	$f_5(\tilde{x}^3)$

B. Попълнете таблицата, като с 1 означите принадлежност към съответното множество, а с 0 – непринадлежност:

	T_0	T_1	S	M	L
$f_1(\tilde{x}^3)$					
$f_2(\tilde{x}^3)$					
$f_3(\tilde{x}^3)$					
$f_4(\tilde{x}^3)$					
$f_5(\tilde{x}^3)$					

C. Напишете функцията на покритие за горната таблица.

D. Определете всички бази, ако множеството от функции е пълно.

- 3) **{2 т.}** Да се построи детерминиран краен автомат, разпознаващ езика, породен от дадената по-долу граматика :

$$\Gamma = \langle \{ S, A, B, C, D \}, \{ 0, 1 \}, S, \{ S \rightarrow 0A \mid 1C, A \rightarrow 1A \mid 1B, B \rightarrow 0A \mid 0, \\ C \rightarrow 0C \mid 0D, D \rightarrow 1C \mid 1 \} \rangle$$

- 4) **{2 т.}** Да се състави командна процедура на езика на командния интерпретатор shell (bash) за Linux с два параметъра – имена на директории, която копира всички файлове от първата директория, които не са директории и са достъпни за четене, във втората.

- 5) **{2 т.}** Като параметри на командна процедура на езика на командния интерпретатор shell (bash) за Linux са зададени различни идентификатори на потребители на системата. Да се състави последователност от команди, която извежда за всеки от зададените потребители броя на стартираните от него сесии на стандартния изход.

6) {3 т.} Даден е следният фрагмент от програма:

```
int fd;  
close ( 1 );  
fd = creat ( "new_file", 0777 );  
write ( fd, "HELLO\n", sizeof("HELLO\n" ) );  
write ( 1, "HELLO\n", sizeof("HELLO\n" ) );
```

Като резултат от изпълнението на дадената поредица от оператори:

- a) на терминала ще се изведе низът "HELLO" и във файла "new_file" ще се запише низът "HELLO"
- b) във файла "new_file" ще се запише два пъти низът "HELLO"
- c) на терминала ще се изведе два пъти низът "HELLO"

7) {3 т.} Даден е следният фрагмент от програма:

```
int pid, k=0, status;  
if ( (pid = fork() ) == 0 ) k++;  
else wait( &status);  
++k;  
printf( " Stoinostta na k = %d;", k );
```

Като резултат от изпълнението на дадената поредица от оператори на стандартния изход ще се изведе:

- a) Stoinostta na k =2; Stoinostta na k =1;
- b) Stoinostta na k =2;
- c) Stoinostta na k =1;
- d) Stoinostta na k =1; Stoinostta na k =2;

- 8) **{2 т.}** Попълнете липсващите изрази в дефиницията на функцията rev при условие, че тя действа аналогично на вградената функция reverse (т.е. обръща реда на елементите на най-високо ниво на даден списък), като при това генерира линеен итеративен процес.

```
(define (rev l)
  (define (rev1 lst res)
    (if (null? lst)
        res
        (rev1 .....)))
  (rev1 l .....))
```

- 9) **{3 т.}** Напишете оценката на всеки от следващите изрази на езика Scheme:

```
(cadr (cдар '((a (b (c)) a) (d (e f) ((g)))))) → .....
(car (map list (map car '((a b) (c d)))))) → .....
(map (lambda (x) (cons x (map (lambda (x) (* x x)) '(2 3 4)))) '(1 2 3)) → .....
```

- 10) **{3 т.}** Нека са дадени (т.е. оценени от интерпретатора на Scheme) следните изрази:

```
(define make-mystery
  (lambda (the-cnt)
    (lambda (x)
      (cond ((and (number? x) (even? x))
             (set! the-cnt (/ x 2)) the-cnt)
            ((and (number? x) (odd? x))
             (set! the-cnt (+ the-cnt x)) the-cnt)
            (else (set! the-cnt (* the-cnt 10)) the-cnt))))
  (define mystery (make-mystery 1))
```

Напишете оценките на следващите изрази при условие, че тези изрази се оценяват точно в реда (1), (2), (3), (4):

```
(1) (mystery '()) → .....
(2) (mystery 1) → .....
(3) (mystery 2) → .....
(4) (mystery 'a) → .....
```

- 11) **{2 т.}** Нека е дадена (т.е. оценена от интерпретатора на Scheme) следната дефиниция:

```
(define (f l)
  (cond ((null? l) '())
        ((atom? (car l)) (cons (car l) (f (cdr l))))
        (else (append (f (car l)) (f (cdr l))))))
```

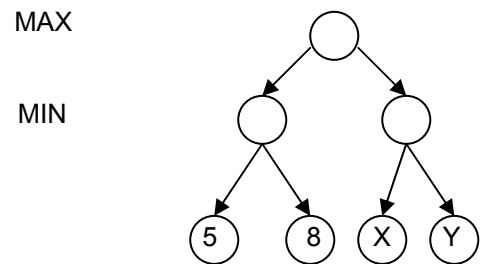
Напишете оценката на следващия израз:

```
(f '((2 (3 (4))) 5 ((6 (7)) 8))) → .....
```

12) {2 т.} Даден е ориентиран граф, представен чрез поредица от факти на Prolog от вида arch(<node1>, <node2>), всеки от които означава, че в графа съществува дъга с начало <node1> и край <node2>. Допишете липсващите аргументи в дефинициите на предикатите breadth_first и extend при условие, че с помощта на предиката breadth_first_search се решава задачата за търсене на път Path от даден начален възел Start да целевия възел Goal, като се реализира алгоритъмът за търсене в широчина:

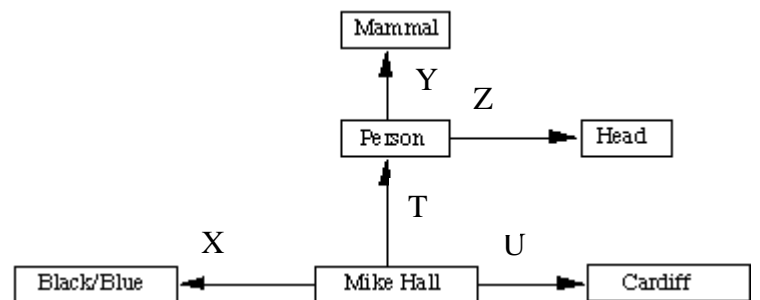
```
breadth_first_search(Start,Goal,Path):-
    breadth_first([[Start]],Goal,L, reverse(L,Path).
breadth_first([[Goal|Path]|_],Goal,[Goal|Path]).
breadth_first([Path|LP],Goal,FinalPath):-
    extend(Path,NewPaths),
    append( ..... , ..... ),
    breadth_first( ..... ,Goal,FinalPath).
extend([Node|Path],NewPaths):-
    findall([NewNode,Node|Path], ..... ,NewPaths).
```

13) {1 т.} На фигурата е дадено игрово дърво. При какви оценки на възела X може да бъде отсечен възелът Y?



14) {2 т.} На фигурата е дадена семантична мрежа. Определете връзките между възлите и попълнете таблицата (запишете знак "X" в подходящите клетки).

	X	Y	Z	T	U
<i>team</i>					
<i>instance</i>					
<i>subclass</i>					
<i>has_part</i>					
<i>team_colors</i>					



15) {3 т.} Фреймовете в дадена фреймова система имат следната структура:

```
(<frame-name> (<slot-1> (<facet-11> <value-11>)
               (<facet-12> <value-12>)
               ... )
(<slot-2> (<facet-21> <value-21>)
          (<facet-22> <value-22>)
          ... ) ... )
```

В базата от знания на системата са дефинирани следните фреймове:

```
(mammal (no_of_legs (default 4)) (cover (value hair)))
(elephant (isa (value mammal)) (has_part (value trunk))
          (cover (default skin)) (color (default gray)))
(clyde (isa (value elephant)) (color (value white)))
```

A. Какъв ще бъде резултатът от Z-търсенето в посочената база от фреймове за слота cover на фрейма clyde?

B. Какъв ще бъде резултатът от N-търсенето в посочената база от фреймове за слота cover на фрейма clyde?

16) {2 т.} Какъв ще бъде резултатът от изпълнението на следния програмен фрагмент:

```
int A[100], j ;
for ( j = 1 ; j <= 10 ; j++ ) A[ j ] = j ;
for ( j = 1 ; j <= 7 ; j++ ) A[ j + 3 ] = A [ j ] ;
for ( j = 1 ; j <= 5 ; j++ ) A[ 10 ] += A [ j+5 ] ;
printf ( " %i ", A[ 10 ] ) ;
```

- a) 10 b) 20 c) 30 d) 40

17) {3 т.} Програмният текст

```
int days ( int month )
{ int result = 28;
  const int d30 = ..... ;
  const int d31 = ..... ;
  switch ( month )
  { default :
    case 4: case 6: case 9: case 11: result += d30;
    case 2: ;
  }
  return (result) ;
}
```

реализира целочислена функция на целочислен аргумент, в съответствие с таблицата:

month	1	2	3	4	5	6	7	8	9	10	11	12
days	31	28	31	30	31	30	31	31	30	31	30	31

Допишете липсващите стойности на именуваните константи.

18) {4 т.} Годишният календар е разпределен в:

```
две полугодия                      h ∈ H = {1, 2} ,
всяко от които с по две тримесечия      t ∈ T = {1, 2} ,
всяко от които с по три месеца              m ∈ M = {1, 2, 3}.
```

При тези условия тройката входни данни (h, t, m) ∈ H × T × M посочва един конкретен календарен месец от една невисокосна година.

Допишете съответните оператори в следващата функция на езика C така, че тя да може да изготвя справка за броя на календарните дни в посочения месец.

```
int days (int h, int t, int m)
{ int D [2][2][3] = .....;
  int *p;
  p = & D[0][0][0] + .....;
  return *p;
}
```

19) {3 т.} В програмния текст на функцията

```
void print_sequence ( int n )
{ ..... // a
  if ( n )
  { ..... // b
    print_sequence ( n-1 );
    ..... // c
  }
  ..... // d
}
```

е пропуснат операторът printf ("%i ", n); .

Добавете липсващия оператор в подходящата измежду позициите //a , //b , //c или //d ,

така че при изпълнение на обръщението: print_sequence (10);

на екрана да бъде изведен текстът : 1 2 3 4 5 6 7 8 9 10

20) **{2 т.}** Обяснете какво прави (какъв резултат извежда) следната програма:

```
#include <stdio.h>
void niz(char *);
void main()
{ char str[40];
  printf("\nМоля въведете низа:\n");
  scanf("%s", str);
  niz(str);
}
void niz(char *s)
{ if (*s != NULL)
  { niz(++s);
    putchar(*--s);
  }
}
```

21) **{2 т.}** Функцията dequeuee изключва елемент от последователна опашка. Попълнете липсващия оператор на мястото на многоточието в следния фрагмент от програма:

```
#define M 100
struct queue
{ int f; // номер на фиктивен елемент преди началото на опашката
  int r; // номер на елемента в края на опашката
  float q_array[M];
};
float dequeuee(queue *q)
{ if (q->r == q->f)
  { cout << "\nПразна опашка\n";
    exit(1); }
.....
  return q->q_array[q->f-1];
}
```

22) **{2 т.}** Функцията dequeuee изключва елемент от свързана опашка. Попълнете липсващите оператори на местата на двете многоточия в следния фрагмент от програма:

```
struct queue_el
{ double info;
  queue_el *link;
};
// формален параметър f – указател към фиктивен елемент
// преди началото на опашката
// формален параметър r – указател към елемента в края на опашката
double dequeuee (queue_el *f, queue_el *&r)
{ double x;
  queue_el *p = f->link;
  if (p == NULL)
  { cout << "\nОпашката е празна\n";
    exit(1); }
.....
  x = p->info;
  delete p;
  if (f->link == NULL)
.....
  return x;
}
```

23) **{2 т.}** Функцията search търси елемент по ключ в свързан списък и го актуализира. Попълнете липсващите оператори на местата на двете многоточия в следния фрагмент от програма:

```
struct student
{ long nomer; // факултетен номер
```

```

float sr_uspeh; // среден успех
student *next;
};
// first – указател към началото на свързания списък
// key – ключ. факултетният номер се разглежда като ключ
// value – стойност, която ще актуализира средния успех
void search(student *first, long key, float value)
{ student *ptr = first;
  while (ptr)
    if (ptr->nomer == key)
      .....
    else
      .....
  if(ptr)
    ptr->sr_uspeh = value;
  else
    cout << "\nНяма елемент с указания ключ";
}

```

- 24) **{2 т.}** Напишете какъв резултат извежда следващата програма. За по-голяма прегледност означете всеки един изведен интервал с един знак :

```

#include <iostream.h>
#include <iomanip.h>
void main()
{ int x = 123;
  cout << setw(6) << x << endl << setw(6);
  cout << setiosflags(ios::left) << x << endl;
  cout << resetiosflags(ios::left) << x << endl;
  cout.fill('*');
  cout << setw(6) << x << endl;
  cout.setf(ios::internal, ios::adjustfield);
  cout << setiosflags(ios::showbase) << setw(8) << hex << x << endl;
  cout << setiosflags(ios::showpos) << setw(8) << dec << x << endl;
  cout << 9.9000 << endl;
  cout.setf(ios::showpoint);
  cout << 9.9000 << endl;
}

```

- 25) **{2 т.}** Напишете какъв резултат извежда следващата програма, ако въведеното изречение от клавиатурата е: Това е проверка на width ^Z

```

#include <iostream.h>
void main()
{ int w = 4;
  char string[10];
  cout << "Въведете изречение: " << endl;
  cin.width(5);
  while (cin >> string)
  { cout.width(w++);
    cout << string << endl;
    cin.width(5);
  }
}

```

- 26) **{2 т.}** Да се открие и коригира грешката в следния фрагмент от дефинирането на клас Array и на конструктора за копие:

```

class Array {
public:
  ...
  Array(const Array &);

```



```
...
private:
    int *ptr; // указател към първия елемент на масива
    int size; // брой на елементите на масива
};
Array::Array(const Array &init)
{ size = init.size;
  ptr = new int[size];
  assert(ptr!=0);
}
```

- 27) **{2 т.}** Да се открие, обясни и коригира грешката в следния фрагмент от дефинирането на клас String и предефиниращата функция на операцията за индексване:

```
class String {
public:
    ...
    char operator[](int);
    ...
private:
    char *sPtr; // указател към началото на низа
    int length; // дължина на низа
};
char String::operator[](int subscript)
{ assert (subscript >= 0 && subscript < length);
  return sPtr[subscript];
}
```

- 28) {2 т.} Нека φ и ψ са затворени формули. Кои от следните твърдения са верни?
- формулата $(\varphi \& \psi)$ е изпълнима \square φ е изпълнима и ψ е изпълнима
 - формулата $(\varphi \vee \psi)$ е изпълнима \square φ е изпълнима и ψ е изпълнима
 - $\models (\varphi \& \psi) \square \models \varphi$ и $\models \psi$
 - $\models (\varphi \vee \psi) \square \models \varphi$ или $\models \psi$
- 29) {2 т.} Нека $\varphi = \forall X (p(X) \rightarrow \neg \exists Y (p(Y) \vee q(X, Y)))$. Кое от следните твърдения е вярно?
- съществува универсална формула, еквивалентна на φ
 - съществува екзистенциална формула, еквивалентна на φ
 - нито а), нито б) е вярно
- 30) {2 т.} Нека P е следната програма на Пролог:
- ```

p([],Z,Z).
p([X|Y],Z,T):- p(Y,[X|Z],T).
?- p([a,b,c],[],T).

```
- Резултатът е:
- $T = [a,b,c]$
  - $T = [b,c,a]$
  - $T = [c,b,a]$
  - No
- 31) {3 т.} Нека  $P$  е следната логическа програма:
- ```

p(0).
p(f(f(f(X)))):- p(X).

```
- Кое от следните твърдения за най-малкия ербранов модел M_p на P е в сила?
- $M_p = \{p(f^n(0)) : n \in \mathbb{N}\}$
 - $M_p = \{p(f^{3n}(0)) : n \in \mathbb{N}\}$
 - $M_p = \{p(f^{3n+1}(0)) : n \in \mathbb{N}\}$
 - $M_p = \emptyset$
- 32) {2 т.} Нека P е условие от тип тотална коректност. Кое от следните твърдения е вярно?
- P е непрекъснато и правилото на Скот е приложимо за него
 - P е непрекъснато и правилото на Скот е неприложимо за него
 - P не е непрекъснато и правилото на Скот е приложимо за него
 - P не е непрекъснато и правилото на Скот е неприложимо за него
- 33) {3 т.} Нека е дадена рекурсивната програма R :
- ```

F(X), where
F(X) = if X=0 then 2 else 3*F(X).

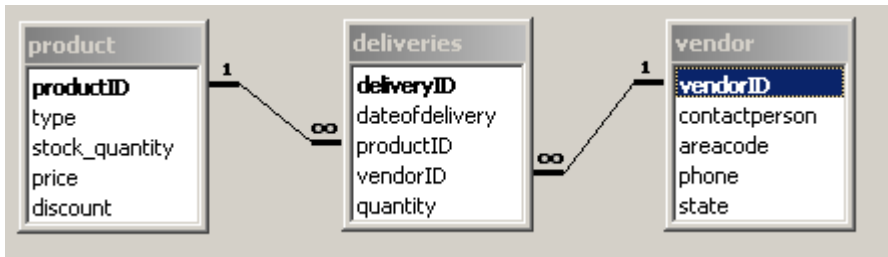
```
- Кое от следните твърдения не е вярно?
- всички стойности на  $D_{\text{тв}}(R)$  са четни
  - всички стойности на  $D_{\text{тв}}(R)$  са прости числа
  - има стойност на  $D_{\text{тв}}(R)$ , която се дели на 3
  - има стойност на  $D_{\text{тв}}(R)$ , която е четна
- 34) {3 т.} Кое от следните свойства на едноместните частични функции не е непрекъснато:
- $P_1(f) = (\forall x)(!f(x) \rightarrow f(x) > x + 17)$
  - $P_2(f) = (\forall x)(f(x) > f(f(x)) + 3)$
  - $P_3(f) = (P_1(f) \& P_2(f))$
  - $P_4(f) = (\exists x)(\forall y) (!f(y) \rightarrow f(y) < x)$
- 35) {3 т.} Нека  $\Gamma(f)$  е оператор от тип  $(1 \rightarrow 1)$ , определен чрез равенството:  $\Gamma(f)(x) > f(f(3*x))$ . Тогава кое от следните твърдения е в сила?
- $\Gamma$  няма неподвижни точки
  - $\Gamma$  има само една неподвижна точка
  - $\Gamma$  има безброй много неподвижни точки
  - $\Gamma$  няма най-малка неподвижна точка



**Задача 2**

Фирма предлага на пазара продукти, доставяни от различни доставчици. За всеки продукт се поддържа следната информация: уникален код, тип на продукта, колко бройки са в наличност в склада (stock\_quantity), цена за единична бройка, отстъпка. Всеки продукт се доставя от един доставчик. Всеки доставчик може да доставя нито един, един или повече продукти. За доставчиците се съхранява информация за лице за контакти, код на областта, телефон, щат. За доставките се съхранява следната информация: уникален код, дата на доставка, кой продукт от кой доставчик и в какво количество е доставен.

Базата **delivery** съдържа три таблици:



В таблица product има следните данни:

| productID | type      | stock_quantity | price  | discount |
|-----------|-----------|----------------|--------|----------|
| A102      | food      | 25             | 100.50 | 0.05     |
| A104      | food      | 8              | 53.75  | 0.1      |
| A250      | food      | 40             | 75.50  | 0.05     |
| B349      | drink     | 80             | 34.00  | 0        |
| B401      | drink     | 53             | 25.00  | 0        |
| B567      | drink     | 8              | 15.00  | 0.1      |
| B709      | drink     | 6              | 65.00  | 0.07     |
| C444      | beverage  | 84             | 8.00   | 0.1      |
| C458      | beverage  | 28             | 20.00  | 0.05     |
| D476      | vegetable | 65             | 27.50  | 0.15     |
| D897      | vegetable | 45             | 89.50  | 0.12     |
| D900      | vegetable | 38             | 8.00   | 0.08     |

В таблица vendor има следните данни:

| vendorID | contactperson | areacode | phone    | state |
|----------|---------------|----------|----------|-------|
| 21222    | Smitson       | 972      | 666-8888 | TX    |
| 21232    | Andersen      | 817      | 222-7777 | TX    |
| 22000    | Antonio       | 304      | 888-9999 | FL    |
| 23002    | Settenen      | 302      | 111-2222 | TX    |
| 25489    | Boyce         | 212      | 333-4444 | FL    |

В таблица deliveries има следните данни:

| deliveryID | dateofdelivery | productID | vendorID | quantity |
|------------|----------------|-----------|----------|----------|
| 120        | 9/6/2003       | A250      | 21222    | 20       |
| 121        | 10/6/2003      | A104      | 25489    | 10       |
| 122        | 10/6/2003      | B709      | 25489    | 50       |
| 123        | 11/6/2003      | C444      | 21232    | 14       |
| 124        | 11/6/2003      | D897      | 25489    | 60       |

- 1) **{1 т.}** За базата от данни delivery кой от следните оператори ще изведе кодовете, цените и отстъпките на продуктите, подредени по цена от най-скъпия до най-евтиния продукт?
- a) 

```
SELECT productID, price, discount
FROM product
ORDER BY price DESC;
```
  - b) 

```
SELECT productID, price, discount
FROM product
ORDER BY price ASC;
```
  - c) 

```
SELECT productID, price, discount
FROM product
SORT BY price DESC;
```
  - d) 

```
SELECT productID, price, discount
FROM product
GROUP BY price DESC;
```
- 2) **{1 т.}** За базата от данни delivery кой от следните оператори ще изведе кодовете на тези продукти, чиято цена е по-ниска от средната цена за продуктите?
- a) 

```
SELECT productID
FROM product
WHERE price <
 (SELECT AVG(price)
 FROM product);
```
  - b) 

```
SELECT productID
FROM product
WHERE price < AVG(price);
```
  - c) 

```
SELECT productID
FROM product
WHERE price <
 (SELECT AVERAGE(price)
 FROM product);
```
  - d) 

```
SELECT productID
FROM product
WHERE price < AVERAGE(price);
```
- 3) **{1 т.}** За базата от данни delivery кой от следните оператори ще изведе най-високата цена за всеки от типовете продукти, за който средната цена за типа е по-голяма от 30?
- a) 

```
SELECT type, MAX(price)
FROM product
HAVING SUM(price)>30
GROUP BY type;
```
  - b) 

```
SELECT type, MAX(price)
FROM product
GROUP BY type
WHERE AVG(price)>30;
```
  - c) 

```
SELECT type, MAX(price)
FROM product
GROUP BY type
HAVING AVG(price)>30;
```
  - d) 

```
SELECT type, MAX(price)
FROM product
WHERE AVG(price)>30
GROUP BY type;
```

- 4) {1 т.} За базата от данни delivery кой от следните оператори ще изведе най-голямото доставяно количество (quantity) за всеки от доставчиците, за които броят на доставяните продукти е по-голям от 2?
- a) SELECT vendorID, MAX(quantity)  
FROM deliveries  
HAVING COUNT(productID) > 2;
  - b) SELECT vendorID, MAX(quantity)  
FROM deliveries  
GROUP BY vendorID  
HAVING SUM(productID) > 2;
  - c) SELECT vendorID, MAX(quantity)  
FROM deliveries  
GROUP BY vendorID  
WHERE COUNT(productID) > 2;
  - d) SELECT vendorID, MAX(quantity)  
FROM deliveries  
GROUP BY vendorID  
HAVING COUNT(productID) > 2;
- 5) {1 т.} За базата от данни delivery кой от следните оператори ще изведе средната цена на продуктите от тип beverage?
- a) SELECT AVG(price)  
FROM product  
GROUP BY product.type;
  - b) SELECT AVG(price)  
FROM product  
GROUP BY product.type  
WHERE type='beverage';
  - c) SELECT AVG(price)  
FROM product  
GROUP BY product.type  
HAVING type='beverage';
  - d) SELECT AVG(price)  
FROM product  
GROUP BY product.price  
HAVING type='beverage';