

ИЗПИТ ПО ФУНКЦИОНАЛНО ПРОГРАМИРАНЕ
януарска сесия, 2004/05 учебна година (вариант 2)

име фак. № група

1. Какъв ще бъде резултатът от изпълнението на следната програма на езика Scheme:

```
(define (f x y)
  (if (= x 1) (- x) y))
(define (g) ((lambda () 1)))
(define (h) (h))
(f (g) (h))
```

ако реализацията на интерпретатора се основава на нормалния модел на оценяване?

- а) съобщение за грешка
- б) изпълнението няма да завърши
- в) 1
- г) -1

2. Какъв тип изчислителен процес генерира следната програма на езика Scheme:

```
(define (f x n)
  (define (square x) (* x x))
  (cond ((= n 0) 1)
        ((even? n) (square (f x (/ n 2))))
        (else (* x (f x (- n 1))))))
```

- а) линеен рекурсивен процес
- б) линеен итеративен процес
- в) логаритмичен рекурсивен процес
- г) логаритмичен итеративен процес

3. Напишете оценката на всеки от следващите изрази на езика Scheme:

```
(car (cdr '(a (b c) d))) → .....
(cdr (car '(((a b) (c d)) (e f)))) → .....
(cdr (cdr '((a b) ((c d)))))) → .....
```

4. 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 следния израз:

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

5. Какво е предназначението на следната функция на езика Scheme:

```
(define (f l1 l2)
  (cond ((null? l2) l1)
        ((member (car l2) l1) (f l1 (cdr l2)))
        (else (cons (car l2) (f l1 (cdr l2))))))
```

- а) намира сечението $l1 \cap l2$
- б) намира обединението $l1 \cup l2$
- в) намира разликата $l1 \setminus l2$
- г) намира разликата $l2 \setminus l1$

6. Нека е дадено (т.е. оценено от интерпретатора на Scheme) следното множество от дефиниции:

```
(define (function l)
  (if (null? l)
      '()
      (cons (car l) (function (delete-all (car l) (cdr l))))))
(define (delete-all s l)
  (cond ((null? l) '())
        ((equal? (car l) s) (delete-all s (cdr l)))
        (else (cons (car l) (delete-all s (cdr l))))))
```

Напишете оценката на следния израз:

```
(function '(a s d f c a f e d a)) → .....
```

7. Нека са дадени (т.е. оценени от интерпретатора на Scheme) следните изрази:

```
(define make-mystery
  (lambda (the-list)
    (lambda (x)
      (cond ((null? x) the-list)
            ((atom? x) (set! the-list (cons x the-list)))
            (else (set! the-list (append x the-list))))))
(define mystery (make-mystery '(a)))
```

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

- (1) `(mystery '())` →
- (2) `(mystery 'b)` →
- (3) `(mystery '(c d))` →

- 8. Обяснете същността на модела на заместването при оценяване на обръщения към съставни процедури в езика Scheme.

- 9. Каква е областта на действие на локалните имена, дефинирани с помощта на специалната форма `letrec`?

- 10. Дефинирайте понятието “списък” в езика Scheme.

- 11. Обяснете действието на вградената функция за проверка на равенство `equal?` в езика Scheme.

- 12. Обяснете действието на вградената функция `map` в езика Scheme. Дайте примери.

- 13. Как се извършва прилагането на дадена съставна процедура към съответните аргументи съгласно модела на средите?

- 14. Обяснете действието на вградената функция `set-cdr!` в езика Scheme. Дайте примери.

- 15. Анализирайте предназначението и обяснете действието на специалната форма `cons-stream` в езика Scheme.

- 16. Обяснете действието на специалната форма `macro` в езика Scheme. Дайте пример.