

<b>Started on</b>	Tuesday, 28 March 2023, 2:07 PM
<b>State</b>	Finished
<b>Completed on</b>	Tuesday, 28 March 2023, 2:16 PM
<b>Time taken</b>	9 mins 54 secs
<b>Marks</b>	1.00/4.00
<b>Grade</b>	<b>2.50</b> out of 10.00 ( <b>25%</b> )

**Question 1**

Incorrect

Mark 0.00 out of 1.00

Ce rezultate vor întoarce apelurile de funcții următoare?

```
(foldr + 0 '(1 2 3))
```

```
(foldl + 0 '(1 2 3))
```

```
(apply + 1 2 3)
```

Select one:

- ☐ a. apelurile de foldr și foldl vor întoarce  
↪ iar ultima va da eroare pentru că nu  
↪ fost dat ca argument un acumulator
- ☐ b. toate vor întoarce rezultatul 6
- ☐ c. apelurile de foldr și foldl vor întoarce  
↪ iar ultima va da eroare pentru că ap  
↪ nu are listă ca argument final
- ☒ d. apelurile de foldr și foldl vor întoarce  
↪ '(6), iar apelul de apply va întoarc

Răspunsul dumneavoastră este incorect.

The correct answer is:

apelurile de foldr și foldl vor întoarce 6,  
↪ iar ultima va da eroare pentru că apply  
↪ nu are listă ca argument final

**Question 2**

Incorrect

Mark 0.00 out of 1.00

Fie următoarea funcție:

```
(define (f x y)  
  (+ x y))
```

Care dintre următoarele expresii NU va genera eroare?

Select one:

- ☐ a. (((f) 1) 2)
- ☐ b. (f 1)
- ☐ c. f
- ☒ d. ((f 1) 2) ✖

Răspunsul dumneavoastră este incorrect.

The correct answer is: f

**Question 3**

Incorrect

Mark 0.00 out of 1.00

Se dă următorul program Racket:

```
(define L '(1 2 3 4 5 6 7 8))
(foldl (lambda (elem acc)
        (if (odd? elem) acc (cons
            elem acc)))
    '() L)
(foldr (lambda (elem acc)
        (if (even? elem) (cons elem
            acc) acc))
    '() L)
(filter odd? L)
(map (lambda (x) (if (even? x) x)) L)
)
```

Aplicația cărei funcții se va evalua la lista '(2 4 6 8)?

Select one:

- ☐ a. foldl
- ☐ b. filter
- ☒ c. map
- ☐ d. foldr

✖

Răspunsul dumneavoastră este incorrect.

**foldr**

The correct answer is:

**Question 4**

Correct

Mark 1.00 out of 1.00

Ce calculează funcțiile  $f$  și  $g$ ?

```
(define (g L)
  (foldr (lambda (x acc) (cons L
    ↪ acc)) '() L))
```

```
(define (f L)
  (apply *
    (map length (g L))))
```

Obs:  $n$  este lungimea listei  $L$  dată ca parametru.

- ☐ a.  $g$ : o listă formată din toate listele sufix ale listei  $L$   
 $f$ :  $n!$  ( $n$  factorial)
- ☒ b.  $g$ : o listă formată din  $n$  copii ale listei  $L$  ✓  
 $f$ :  $n^n$  ( $n$  la puterea  $n$ )
- ☐ c.  $g$ : o listă formată din elementele listei  $L$ , copiate de  $n$  ori  
 $f$ :  $n \cdot n$
- ☐ d.  $g$ : o listă formată din toate listele prefix ale listei  $L$   
 $f$ :  $n!$  ( $n$  factorial)

Răspunsul dumneavoastră este corect.

The correct answer is:

$g$ : o listă formată din  $n$  copii ale listei  $L$   
 $f$ :  $n^n$  ( $n$  la puterea  $n$ )