

PARADIGMATA PROGRAMOVÁNÍ I

zápočtový test

1. Co je s-výraz (symbolický výraz)?
Proč mluvíme o s-výrazech v souvislosti s jazykem Scheme?
Určete, které z následujících výrazů jsou s-výrazy:
 $+$, $3/4$, $(+ 1 2)$, $(1 + 2)$
2. Popište vyhodnocování v jazyce Scheme (vyhodnocování výrazu e v prostředí \mathcal{P}).
3. Vysvětlete jaký je rozdíl mezi procedurou a speciální formou.
Uveďte 3 příklady od každého.
4. Popište vznik a aplikaci uživatelsky definované procedury.
5. Napište kód procedury `summax2`, která přijímá 3 argumenty a vrací součet 2 větších.
6. Jaký je rozdíl mezi statickým a dynamickým rozsahem platnosti? Napište kód ve Scheme, který by se vyhodnotil na jinou hodnotu při statickém a dynamickém rozsahu platnosti.
Který z nich Scheme používá?
7. Uvažujte následující program:

```
(define kons (lambda (a b) (lambda (p) (if p a b))))  
(define kar (lambda (c) (c #t)))  
(car (kons 7 8))
```

Podrobně popište, jak se vyhodnotí poslední výraz. Dále: zakreslete hierarchii prostředí, která při jeho vyhodnocování vzniká.
8. Napište seznam, jehož první prvek není symbol a při jeho vyhodnocování nenastane chyba. Pokud takový výraz neexistuje, vysvětlete proč.

9. Napište na co se vyhodnotí následující výrazy ve Scheme. Pokud při vyhodnocování nastane chyba, uveďte proč:

(a) `(+ 1 (* 2 3))`

(b) `(+ 1 (2 * 3))`

(c) `(and 13 (/ 7 0))`

(d) `(let ((x 10) (y x)) (* x y))`

(e) `(let ((x (+ 5 5))) (* x x) (+ x x))`

(f) `(let* ((x 10) (y (lambda () x)) (x 5))
 (+ x (y)))`

(g) `(caar (cons (cons 1 2) 3))`

(h) `(cons '(cons . cons) car)`

PARADIGMATA PROGRAMOVÁNÍ I

zápočtový test

1. Co je s-výraz (symbolický výraz)?
Proč mluvíme o s-výrazech v souvislosti s jazykem Scheme?
Určete, které z následujících výrazů jsou s-výrazy:
 $+$, $3/4$, $(+ 1 2)$, $(1 + 2)$
2. Popište vyhodnocování v jazyce Scheme (vyhodnocování výrazu e v prostředí \mathcal{P}).
3. Vysvětlete jaký je rozdíl mezi procedurou a speciální formou.
Uveďte 3 příklady od každého.
4. Popište vznik a aplikaci uživatelsky definované procedury.
5. Napište kód procedury `summax2`, která přijímá 3 argumenty a vrací součet 2 větších.
6. Jaký je rozdíl mezi statickým a dynamickým rozsahem platnosti? Napište kód ve Scheme, který by se vyhodnotil na jinou hodnotu při statickém a dynamickém rozsahu platnosti.
Který z nich Scheme používá?
7. Uvažujte následující program:

```
(define kons (lambda (a b) (lambda (p) (if  
p a b))))  
(define kar (lambda (c) (c #t)))  
(car (kons 7 8))
```

Podrobně popište, jak se vyhodnotí poslední výraz. Dále: zakreslete hierarchii prostředí, která při jeho vyhodnocování vzniká.
8. Napište seznam, jehož první prvek není symbol a při jeho vyhodnocování nenastane chyba. Pokud takový výraz neexistuje, vysvětlete proč.

9. Napište na co se vyhodnotí následující výrazy ve Scheme. Pokud při vyhodnocování nastane chyba, uveďte proč:

(a) `(+ 1 (* 2 3))`

(b) `(+ 1 (2 * 3))`

(c) `(and 13 (/ 7 0))`

(d) `(let ((x 10) (y x)) (* x y))`

(e) `(let ((x (+ 5 5))) (* x x) (+ x x))`

(f) `(let* ((x 10) (y (lambda () x)) (x 5))
 (+ x (y)))`

(g) `(caar (cons (cons 1 2) 3))`

(h) `(cons '(cons . cons) car)`

PARADIGMATA PROGRAMOVÁNÍ I

zápočtový test

1. Co je s-výraz (symbolický výraz)?
Proč mluvíme o s-výrazech v souvislosti s jazykem Scheme?
Určete, které z následujících výrazů jsou s-výrazy:
 $+$, $3/4$, $(+ 1 2)$, $(1 + 2)$
2. Popište vyhodnocování v jazyce Scheme (vyhodnocování výrazu e v prostředí \mathcal{P}).
3. Vysvětlete jaký je rozdíl mezi procedurou a speciální formou.
Uveďte 3 příklady od každého.
4. Popište vznik a aplikaci uživatelsky definované procedury.
5. Napište kód procedury `summax2`, která přijímá 3 argumenty a vrací součet 2 větších.
6. Jaký je rozdíl mezi statickým a dynamickým rozsahem platnosti? Napište kód ve Scheme, který by se vyhodnotil na jinou hodnotu při statickém a dynamickém rozsahu platnosti.
Který z nich Scheme používá?
7. Uvažujte následující program:

```
(define kons (lambda (a b) (lambda (p) (if p a b))))  
(define kar (lambda (c) (c #t)))  
(car (kons 7 8))
```

Podrobně popište, jak se vyhodnotí poslední výraz. Dále: zakreslete hierarchii prostředí, která při jeho vyhodnocování vzniká.
8. Napište seznam, jehož první prvek není symbol a při jeho vyhodnocování nenastane chyba. Pokud takový výraz neexistuje, vysvětlete proč.

9. Napište na co se vyhodnotí následující výrazy ve Scheme. Pokud při vyhodnocování nastane chyba, uveďte proč:

(a) `(+ 1 (* 2 3))`

(b) `(+ 1 (2 * 3))`

(c) `(and 13 (/ 7 0))`

(d) `(let ((x 10) (y x)) (* x y))`

(e) `(let ((x (+ 5 5))) (* x x) (+ x x))`

(f) `(let* ((x 10) (y (lambda () x)) (x 5))
 (+ x (y)))`

(g) `(caar (cons (cons 1 2) 3))`

(h) `(cons '(cons . cons) car)`