Computer Programming

Training problems for M3 2018 term 2

Ted Szylowiec
tedszy@gmail.com

You can find SICP (*Structure and Interpretation of Computer Programs*) online here:

https://sarabander.github.io/sicp/

Download Racket here:

https://racket-lang.org/

Use Racket online at Tio:

https://tio.run/#racket

1 Lambda

1. Use `define` to define a symbol having an integer value.

2. Use `define` to define a symbol having a string value.

3. Use `define` to define a symbol having a boolean value.

4. Define a symbol to have a rational value.

5. Define a symbol to have a float value.

6. Use `define` and `lambda` to define a symbol having a function value.

7. Explain why these give you errors.
(a) `(define "x" 10)`
(b) `(define 10 5)`
(c) `(define #f a)`
(d) `("string-append" "good" "night")`
(e) `(define (f "x") (* x x))`
(f) `(define ("f" x) (* x x))`

8. What is a lambda? Who discovered it? Why is it so interesting in computer science?

9. Give some examples of computer programming languages that have lambda and support lambda-style programming.

10. Practice arrow notation. What is the result?
(a) $(x \to x^2 + 1)(3)$
(b) $(x, y \to 2x + 5y)(3, 7)$

(c) $(x, y, z \to \sqrt{xy} + \sqrt{xz} + \sqrt{yz})(2, 3, 5)$
(d) $(x, y, z \to |xy| + |xz| + |yz|)(-1, 2, -3)$
(e) $(x, y \to x^2 + y^2)((x \to x+1)(2), (x \to x-2)(7))$

11. Write this as a lambda expression: $x \to x^2 + 3x + 1$.

12. Write this as a lambda expression: $x \to x^2$ if x is odd, else x^3. Use Racket's `if` and `odd?` function.

13. Write this as a lambda expression: $x, y \to \sqrt{xy}$. Use Racket's `sqrt` function.

14. Write using lambda: $x, y, z \to \frac{x^2+y^2+z^2}{2}$.

15. The identity function takes x and returns x without any changes: $x \to x$. Write the identity function using lambda.

16. Change lambda expression to arrow (→) notation:

```
(lambda (x y) (+ (* 2 x) (* 3 y)))
```

17. Change lambda expression to arrow notation:

```
(lambda (x y z) (+ (/ (sqrt x))
                   (/ (sqrt y))
                   (/ (sqrt z))))
```

18. What does Racket return?
(a) `> (lambda (x) (* x x))`
(b) `> ((lambda (x) (* x x)) 5)`
(c) `> ((lambda (x y) (+ 1 (* x y))) 6 7)`
(d) `> ((lambda (x) (string-append "happy " x)) "halloween")`
(e) `> ((lambda (x) (string-append x "happy ")) "halloween")`

19. What does Racket return?
(a) `> ((lambda (x y z) (+ x y z)) 10 21 32)`
(b) `> ((lambda (x y z) (+ (/ x) (/ y) (/ z))) 2 3 5)`
(c) `> ((lambda (x y) (* (+ x y) (- x y))) 7 5)`

20. What does this expression return?

```
((lambda (x)
    (* ((lambda (y) (+ (* 2 y) 1)) x)
       ((lambda (y) (- y 1)) x)))
 10)
```

21. Write a lambda-expression that adds the square roots of 3 and 5.

22. Write a lambda expression that finds the harmonic mean of 2, 5 and 7.

23. Write a lambda expression that finds the average of the lengths of these two lists: `(list 'a 'b 'c)` and `(list 1 2 3 4 5)`. Use the `length` function to get the length of a list.

24. Let $f : x \to 5x$ and $g : x \to 2x$. Write a one-line lambda expression that does $f(3) + g(6)$.

25. Change this to lambda-style function definition.

```
(define (f x)
   (+ (* x x) 5)
```

26. Change to lambda-style function definition.

```
(define (f x)
   (if (even? x) (/ x 2) (* x 2)))
```

27. Change to lambda-style definition.

```
(define (g x y)
   (/ (+ x y) 2))
```

28. Change to lambda-style definition.

```
(define (h x y z)
   (expt (* x y z) 1/3))
```

29. Do this computation with a one-shot expression using a lambda and no definitions.

```
(define (f x)
   (+ (* 2 x) 1))
(f 10)
```

30. Do this as a one-line expression using lambda, without definitions.

```
(define (greetings s)
   (string-append "hello there " s))
(greetings "Jim")
```

31. Rewrite this as one expression using lambda and no definitions.

```
(define a 10)
(define b 25)
(define (f x y) (- (* x y) 5))
(f a b)
```

32. Rewrite all this as a one-line expression using lambda.

```
(define s1 "greetings ")
(define s2 "earthman")
(define (F a b)
   (string-append a b ", take me to your leader"))
(F s1 s2)
```

33. Get rid of all symbol definitions and rewrite this program as a one-line expression using lambda.

```
(define a 30)
(define b 40)
(define c 60)
(define (average x y z)
         (/ (+ x y z) 3))
(average a b c)
```

34. Let $f : x \to x^2$ and $g : x \to x + 1$. Write $f(g(5))$ as one expression using two lambdas. Don't use `define` or `compose`.

2 Map and filter

35. What does this expression return?

```
(map (lambda (x) (* x x))
     (list 1 2 3 4 5 6 7))
```

3 Logic