

Problems from “How to Solve Problems Using Scheme” for Assignment 2*

1. Exercise 3 of Section 11.4

Write a recursive procedure (`one-to-n n`) which takes an integer `n` and returns a list of the form `(1 2 ... n)`. For example:

```
(one-to-n 5) ==> (1 2 3 4 5)
```

2. Exercise 5 of Section 11.4

Write a recursive procedure (`my-reverse lst`) which returns a list containing the elements of `lst` in reverse order.

3. Exercise 8 of Section 11.4

Write a recursive procedure (`list-sums lst`) which takes a list `lst` and returns a list where the k^{th} element is the sum of the first k elements of `lst`. For example:

```
(list-sums '(1 4 9 -2 7)) ==> (1 5 14 12 19)
```

4. Exercise 10 of Section 11.4

Write a recursive procedure (`positions lst e`) which returns a list of numbers corresponding to the position of every occurrence of element `e` in the list `lst`. Assume the first element of the list is element 0. For example:

```
(positions '(a b f b a f b b) 'b) ==> (1 3 6 7)
```

6. Exercise 2 of Section 12.4

Write a non-recursive procedure (`distance p q`) using `map` and/or `apply` that returns the euclidean distance between the points `p` and `q` in an arbitrary dimension Euclidean space. For example:

```
(distance '(0 3) '(4 0)) ==> 5  
(distance '(2 -8 7) '(-3 1 9)) ==> 10.488088
```

*Copyright ©2001 Wesley H. Huang. All rights reserved. No part of this material may be reproduced or distributed in any form or by any means including photocopying, recording, or any information storage or retrieval system, without written permission from the author.