CS 496: Homework Assignment 1
Due: January 29th, 11:55pm

1 Assignment Policies

Collaboration Policy. Homework will be done individually: each student must hand in their own answers. It is acceptable for students to collaborate in understanding the material but not in solving the problems or programming. Use of the Internet is allowed, but should not include searching for existing solutions.

Under absolutely no circumstances code can be exchanged between students. Excerpts of code presented in class can be used.

Assignments from previous offerings of the course must not be re-used. Violations will be penalized appropriately.

Late Policy. No late submissions will be allowed without consent from the instructor. If urgent or unusual circumstances prohibit you from submitting a homework assignment in time, please e-mail me.

2 Assignment

Implement the following functions:

1. `seven`: that given any value returns 7.
2. `sign`: that given an integer returns 1 if it is positive, -1 if it is negative and 0 if it is zero.
3. `absolute`: the absolute value function.
4. `andp,orp,notp,xorp`: the standard boolean connectives (you must resort to if-then-else).
5. `dividesBy`: that given two numbers determines if the first is divisible by the second (use remainder).
6. `singleton?`: a predicate that, given a list, returns a boolean indicating whether it has exactly one element. Provide two solutions, one using the `match` construct for pattern matching and another using the predicates `cons?` and `null?`.

7. `swap`: a function that, given a pair, returns the same pair except that its first and second components are interchanged. Use `match`.

8. `app`: a function that, given two arguments, applies the first argument to the second one. Eg. if `succ` is the successor function then

```
> (app succ 2)
3
```

9. `twice`: a function that, given two arguments, applies the first one to the second argument and then again to the result. Eg.

```
> (twice succ 2)
4
```

10. `compose`: a function that, given three arguments, applies the second to the third and then the first to to its result. Eg.

```
> (compose succ succ 3)
4
```

3 Submission instructions

Submit a single file named `HW1.rkt` through Canvas. No report is required. Your grade will be determined as follows:

- You will get 0 if your code does not pass the syntax checker (see button in the menu bar of DrRacket).
- Partial credit may be given for style, comments and readability.