Lab 11: Asynchronous Tasks and Macros

Learning objectives:

- Gain experience writing code using asynchronous tasks
- Gain experience writing macros in Scheme

This labs aims to further prepare you for homework 3, which covers C++ macros and asynchronous tasks.

For all Scheme coding questions, you must use recursion and you may not use mutation (`set!`, `set-car!`, `set-cdr!`, etc.).

Use the following commands to download and unpack the distribution code:

```
$ wget https://eecs390.github.io/lab/lab11/starter-files.tar.gz
$ tar xzf starter-files.tar.gz
```

1. *Asynchronous tasks.* The maximum subarray problem is the task of finding a contiguous subset of an array that maximizes the sum over the subset. While this problem can be solved with dynamic programming, we implement a recursive solution in `max_subarray.cpp`. A serial version is contained in the `max_subarray()` function. Complete the parallel version in `async_max_subarray()`, which should launch an asynchronous task to recurse on one half of the array. The other half should be handled in the existing thread.

   Make sure that the two halves are computed in parallel. You'll need to use `std::async()` to launch a task, save the result as a future, then handle the part of the computation that needs to be done in the current thread before invoking `get()` on the future. Due to C++ ordering semantics, this needs to be done in separate statements.

   Test your implementation as follows:

   ```
   $ g++ -std=c++20 -o max_subarray.exe max_subarray.cpp
   $ ./max_subarray.exe 10000000 8
   ```

   You can use the provided `Makefile` to compare against the expected output.

2. *Scheme macros.* Implement the `my-or` macro in Scheme, which should behave exactly like the or library syntax:

   ```
   > (define x 0)
   > (my-or (= x 0) (/ 3 x))
   #t
   > (define x 1)
   > (my-or (= x 0) (/ 3 x))
   3
   ```

   You may use the built-in `if` form in your solution.

   Write your implementation in `my-or.scm`. To test your implementation, run the included tests:

   ```
   $ plt-r5rs my-or.scm
   ```