Data Structures and Concurrency Exercises

Modifying Operations

- A data structure is accessed by two threads concurrently

- For each of these scenarios, state whether there is a potential data race

- Explain your answers
  - Thread A writes to the first element of a linked list and thread B reads the second element
  - Thread A removes the first element of a single-linked list and thread B reads the second element
  - Thread A removes the second element of a double-linked list and thread B reads the first element

Modifying Operations

- Thread A removes the first element of a single-linked list and thread B removes the second element
- Thread A removes the second element of a double-linked list and thread B removes the first element
- Thread A removes the second element of a double-linked list and thread B removes the fifth element

Basic Thread Safety Guarantee

- When two threads can access the same STL container object, what guarantees do we have?

- Write a program which demonstrates unsafe access to an STL container object by threads

- Alter your program so that it is now thread-safe

Coarse-grained Locking

- What is meant by "coarse-grained" locking?

- What are the advantages of coarse-grained locking?

- What are the disadvantages of coarse-grained locking?

Fine-grained Locking

- What is meant by "fine-grained" locking?

- What are the advantages of fine-grained locking?

- What are the disadvantages of fine-grained locking?