Data Races Solutions

Data Races 1

- In the following code sample, func1 and func2 are run as concurrent threads

```
const int x{5};

int func1() {
    return 2*x;
}

int func2() {
    return 3*x;
}
```

- Can a data race occur? Explain your answer

Data Races 1

- x is const
- No thread can modify x
- There is no possibility of conflicting accesses to x
- The code shown is data-race free
  - Unless a thread dangerously casts away const

Data Races 2

- In the following code sample, func1 and func2 are run as concurrent threads

```
int x{0}, y{0};

void func1() {
    if (x)
        y = 1;
}

void func2() {
    if (y)
        x = 1;
}
```

- Can a data race occur? Explain your answer

Data Races 2

- In func1, x is always 0, and y is never set to 1
- In func2, y is always 0, and x is never set to 1
- There is no possible execution path in which more than one thread tries to modify x or y
- The code is data-race free

Data Races 3

- In the following code sample, func1 and func2 are run as concurrent threads

```
int x{0}, y{0};

void func1() {
   x = 1;
   int r1 = y;
}

void func2() {
   y = 1;
   int r2 = x;
}
```

- Can a data race occur? Explain your answer

Data Races 3

- It is possible that one thread reads y and the other thread concurrently modifies y
- The accesses are not synchronized
- There is a data race