

6 Further Java (ACR)

- (a) Describe the operation of `wait()` and `notifyAll()`. Ensure that your answer explains when locks are acquired and released. [5 marks]
- (b) A *future* is a mechanism to store the eventual result of a computation done in another thread. The idea is that the computation is run asynchronously and the calling thread only blocks if it tries to use a result that hasn't been computed yet. An example program using a future is shown below.

```
Future<String> f = new Future<String>() {
    @Override
    public String execute() {
        // ...long running computation...
        return data;
    };
};

// ...

String result = f.get(); // blocks if execute() unfinished
```

Use `wait()` and `notifyAll()` to provide an implementation of the `Future` class that would work with the example program above. [10 marks]

- (c) Give one potential advantage and one potential disadvantage of using `notify()` instead of `notifyAll()`. [2 marks]
- (d) Would it have been beneficial to use `notify()` instead of `notifyAll()` in your implementation? Justify your answer. [3 marks]