

# COMP 734 Midterm 1

Tue, Oct 15, 2013, 2:00-3:15

## Instructions

1. Please spread out and try and sit in alternate seats.
2. This is a closed book exam.
3. Write on the exam itself.
4. Write on the opposite side of a question page if there is not enough space to solve a problem.
5. There are:
  - 6 numbered pages including this one; there are no blank numbered pages
  - 4 questions
  - 75 possible points. Point values appear in brackets next to each question
6. You have 75 minutes.
7. *Explain all answers.*
8. If you need to make any assumptions to clarify a problem, *write your assumptions down*. Only reasonable assumptions get full credit.

Name (in Capitals)

\_\_\_\_\_

Pledge: I have neither given nor received unauthorized aid on this exam.

(signed) \_\_\_\_\_

Please do not write below

1. \_\_\_\_/10

2. \_\_\_\_/20

3. \_\_\_\_/20

4. \_\_\_\_/25

Total: \_\_\_\_/ 75

Consider a distributed system that supports the following operations on a duplex bound port:

```
Port create (portNumber)
Port connect (host, portNumber);
void async_send (port, byteBuffer)
ByteBuffer send_wait (port, byteBuffer)
ByteBuffer receive_wait (port, byteBuffer)
void reply (port, replyByteBuffer)
```

`create()` is executed by one of the processes, the server, to create a port on its host, and takes as an argument an integer `portNumber` that uniquely identifies the port on the host. It returns an object (handle) identifying the port. `connect()` is called by the other process, the client, to create an object identifying the port. `async_send()` does not block; `receive_wait()` waits for a message from a sender; `send_wait()` waits until the receiver replies using `reply()`; and `reply` is a non blocking call that sends an arbitrary value to the port, which is returned by the last `send_wait()` executed by the peer process. The client and server may be on the same or different machines.

### 1. [10pts.] Bound Port

Define a duplex bound port and motivate it using an example.

## 2. [20pts.] Synchronous vs. Non-Blocking

Using motivating examples, give advantages and disadvantages of supporting `async_send()` and `send_wait()`. Be sure to state your assumptions about the underlying system and make only reasonable ones. You need not give more than one motivating example for each advantage/disadvantage.

### 3. [20pts.] Response-Time Comparison

Consider a pair of communicating processes, S1 and R1, that execute the following code.

```
S1  
for i := 1 to N do  
  async_send (port, message);  
od od
```

```
R1  
for i := 1 to N do  
  receive_wait (port, message)  
end do
```

and another communicating pair, S2 and R2, that execute the following code:

```
S2  
for i := 1 to N do  
  reply ← send_wait (port, message);  
od od
```

```
R2  
for i := 1 to N do  
  receive_wait (port, message)  
  reply(port, reply);  
end do
```

The following four experiments are performed and the times taken by the sending process to execute the loop observed:

- (1) S1 and R1 execute on the same computer. Time taken: 4 seconds.
- (2) S1 and R1 execute on different computers. Time taken: 2 seconds.
- (3) S2 and R2 execute on the same computer. Time taken: 6 seconds.
- (4) S2 and R2 execute on different computers. Time taken: 7 seconds.

Explain why experiment (2) takes less time than (1), (3) takes more time than (1), and (4) takes more time than (3). If you cannot justify any difference, explain why not.

[25pts.] NIO

Outline how you would implement these operations in Java using NIO. In your answer, emphasize the aspects that are different from your solution to assignment 1. If you think certain aspects of these operations cannot be implemented, explain.

