**COMS3200 - ASSIGNMENT 1 (PART A)**

**Jiefeng Hou**

**43034002**

**Task 1:**

Registration Client

Catalog Server

User Server

Loans Server

Checkout Client

CheckIn Client

Query Client

**Task 2:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sending Process** | **Send Primitive** | **Receiving Process** | **Receive Primitive** | **Message Format Names** |
| Registration Client | RPC call | Catalog Server | RPC accept | 1 |
| Catalog Server | RPC reply | Registration Client | RPC call | 2 |
| Registration Client | RPC call | User Server | RPC accept | 3 |
| User Server | RPC reply | Registration Client | RPC call | 4 |
| Query Client | RPC call | Catalog Server | RPC accept | 5 |
| Catalog Server | RPC reply | Query Client | RPC call | 6 |
| Query Client | RPC call | Loans Server | Non-blocking receive | 7 |
| Loans Server | Non-blocking send | Query Client | RPC call | 8 |
| CheckIn Client | RPC call | Loans Server | Non-blocking receive | 9 |
| Loans Server | Non-blocking send | CheckIn Client | RPC call | 10 |
| Checkout Client | RPC call | Loans Server | Non-blocking receive | 11 |
| Loans Server | Non-blocking send | Checkout Client | RPC call | 12 |
| Loans Server | Non-blocking send | Catalog Server | RPC accept | 13 |
| Catalog Server | RPC reply | Loans Server | Non-blocking receive | 14 |
| Loans Server | Non-blocking send | User Server | RPC accept | 15 |
| User Server | RPC reply | Loans Server | Non-blocking receive | 16 |

**Task 3:**

**Registration Client**

The Registration Client sends a request to a Catalog Server (User Server) and waits for the reply before taking any action, we should notice that Registration Client behaves like a typical client, so it should use RPC Call.

**Catalog Server & User Server**

The Catalog Server (User Server) behaves like a typical client, because Catalog Server (User Server) blocks until messages arrive and send reply. Therefore, the RPC server accept should be used to receive messages and RPC reply should be used to reply.

**Query Client**

The Query Client sends a request to a Loan Server (Catalog Server) and waits for the reply before taking any action, we should notice that Query Client behaves like a typical client, so it should use RPC Call.

**CheckIn Client & Checkout Client**

The CheckIn Client (Checkout Client) sends a request to a Loans Server and waits for the reply before taking any action, we should notice that CheckIn Client (Checkout Client) behaves like a typical client, so it should use RPC Call.

**Loans Server**

The Loans Server is more difficult than other processes, because the Loans Server communicates several processes, so in order to enable quicker response and receive, Loans Server use non-blocking primitives should be used (for both sending and receiving).

**Task 4:**

**Message 1:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Item ID | Authors | Title | Additional publication information | Catalog Code |
| Fixed length string (L=12) | Variable-length Array(n) [Variable length string] | Variable length string | Variable length string | Variable length string |
| 12 bytes | n\*(8 ~ 36) + 4 | 4 ~ 36 bytes | 4 ~ 36 bytes | 4 ~ 36 bytes |

Total: 28 ~ 124 + n\*36 bytes (n is number of authors)

Assume that the maximum number of authors is 7, so the total size is: 28 ~ 376 bytes.

**Message 2:**

|  |
| --- |
| Reply message (from Catalog Server to Registration Client) |
| Fixed length string (L=1) |
| 4 bytes |

Total: 4 bytes

Assume that if Registration Client can create/update a catalog in Catalog Server, the Catalog Server will reply “Y”, if not, reply “N”.

**Message 3:**

|  |  |  |  |
| --- | --- | --- | --- |
| User ID | User Name | Phone Numbers (home, work, mobile) | Email Address |
| Fixed length string (L=12) | Variable length string | Variable length string\*3 | Variable length string |
| 12 bytes | 4 ~ 36 bytes | 12 ~ 108 bytes | 4 ~ 36 bytes |

Total: 28 ~ 192 bytes

**Message 4:**

|  |
| --- |
| Reply message (from User Server to Registration Client) |
| Fixed length string (L=1) |
| 4 bytes |

Total: 4 bytes

Assume that if Registration Client can create/update a user in User Server, the User Server will reply “Y”, if not, reply “N”.

**Message 5:**

|  |  |  |  |
| --- | --- | --- | --- |
| Number of request (1 to 2) | Title | Author | Catalog Code |
| Unsigned Integer | Variable length string | Variable length string | Variable length string |
| 4 bytes | 8 ~ 36 bytes | 8 ~ 36 bytes | 8 ~ 36 bytes |

Total: 12 ~ 76 bytes

**Message 6:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Number of matches (0 to 20) | Item ID | Authors | Title | Additional publication information | Catalog Code |
| Unsigned Integer | Fixed length string (L=12) | Variable-length Array(n) [Variable length string] | Variable length string | Variable length string | Variable length string |
| 4 bytes | 12 bytes | n\*(8 ~ 36 ) + 4 | 8 ~ 36 bytes | 8 ~ 36 bytes | 8 ~ 36 bytes |

Total: 4 ~ 20\*(124 + n\*36) bytes (n is number of authors)

Assume that the maximum number of authors is 7, so the total size is: 4 ~ 7520 bytes.

**Message 7:**

1. Search loan status of a particular book, by entering the Item ID

|  |
| --- |
| Item ID |
| Fixed length string (L=12) |
| 12 bytes |

Total: 12 bytes

2. Enter their ID and get a list of their current borrowings and holds

|  |
| --- |
| User ID |
| Fixed length string (L=12) |
| 12 bytes |

Total: 12 bytes

3. Enter user ID and Item ID in order to put that book on hold for one week

|  |  |
| --- | --- |
| User ID | Item ID |
| Fixed length string (L=12) | Fixed length string (L=12) |
| 12 bytes | 12 bytes |

Total: 24 bytes

4. Enter user ID and Item ID in order to renew the loan for a period of four weeks

|  |  |
| --- | --- |
| User ID | Item ID |
| Fixed length string (L=12) | Fixed length string (L=12) |
| 12 bytes | 12 bytes |

Total: 24 bytes

**Message 8:**

1. Get loan status of a particular book

|  |  |  |
| --- | --- | --- |
| Number of reply (1 to 2) | Flag (loan, hold and renewed loan) | Date (yyyy-mm-dd) |
| Unsigned Integer | Fixed length string (L=1) | Fixed length string (L=10) |
| 4 bytes | 4 bytes | 12 bytes |

Total: 20 bytes

Assume that if flag is loan, then reply “L”, if flag is renewed loan, then reply “R”, if flag is hold, then reply “H”.

2. Get a list of their current borrowings and holds

|  |  |  |  |
| --- | --- | --- | --- |
| Number of reply (0 to 3) | Item ID | Flag (loan, hold and renewed loan) | Date (yyyy-mm-dd) |
| Unsigned Integer | Fixed length string (L=12) | Fixed length string (L=1) | Fixed length string (L=10) |
| 4 bytes | 12 bytes | 4 bytes | 12 bytes |

Total: 32 bytes

Assume that if flag is loan, then reply “L”, if flag is renewed loan, then reply “R”, if flag is hold, then reply “H”.

3. Put that book on hold for one week

|  |  |  |  |
| --- | --- | --- | --- |
| Number of reply (3) | Item ID | Flag (loan, hold and renewed loan) | Date (yyyy-mm-dd) |
| Unsigned Integer | Fixed length string (L=12) | Fixed length string (L=1) | Fixed length string (L=10) |
| 4 bytes | 12 bytes | 4 bytes | 12 bytes |

Total: 32 bytes

Assume that if flag is loan, then reply “L”, if flag is renewed loan, then reply “R”, if flag is hold, then reply “H”.

4. Renew the loan for a period of four weeks

|  |  |  |  |
| --- | --- | --- | --- |
| Number of reply (3) | Item ID | Flag (loan, hold and renewed loan) | Date (yyyy-mm-dd) |
| Unsigned Integer | Fixed length string (L=12) | Fixed length string (L=1) | Fixed length string (L=10) |
| 4 bytes | 12 bytes | 4 bytes | 12 bytes |

Total: 32 bytes

Assume that if flag is loan, then reply “L”, if flag is renewed loan, then reply “R”, if flag is hold, then reply “H”.

**Message 9:**

|  |
| --- |
| Item ID |
| Fixed length string (L=12) |
| 12 bytes |

Total: 12 bytes

**Message 10:**

|  |  |
| --- | --- |
| Flag (loan, hold and renewed loan) | Date (yyyy-mm-dd) |
| Fixed length string (L=1) | Fixed length string (L=10) |
| 4 bytes | 12 bytes |

Total: 16 bytes

Assume that if flag is loan, then reply “L”, if flag is renewed loan, then reply “R”, if flag is hold, then reply “H”.

**Message 11:**

|  |  |
| --- | --- |
| User ID | Item ID |
| Fixed length string (L=12) | Fixed length string (L=12) |
| 12 bytes | 12 bytes |

Total: 24 bytes

**Message 12:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Name | Book Name (Title) | Authors | Due Date (yyyy-mm-dd) | Holds |
| Variable length string | Variable length string | Variable-length Array(n) [Variable length string] | Fixed length string (L=10) | Fixed length string (L=1) |
| 4 ~ 36 bytes | 4 ~ 36 bytes | n\*(8 ~ 36 ) + 4 | 12 bytes | 4 bytes |

Total: 28 ~ 92 + n\*36 bytes (n is number of authors)

Assume that the maximum number of authors is 7, so the total size is: 4 ~ 344 bytes.

Assume that if this book is on hold to a different user, the Holds is “Y”, if not, the Holds is “N”.

**Message 13:**

|  |
| --- |
| Item ID |
| Fixed length string (L=12) |
| 12 bytes |

Total: 12 bytes

**Message 14:**

|  |  |
| --- | --- |
| Book Name (Title) | Authors |
| Variable length string | Variable-length Array(n) [Variable length string] |
| 4 ~ 36 bytes | n\*(8 ~ 36) + 4 |

Total: 8 ~ 36 + n\*36 bytes

Assume that the maximum number of authors is 7, so the total size is: 4 ~ 288 bytes.

**Message 15:**

|  |
| --- |
| User ID |
| Fixed length string (L=12) |
| 12 bytes |

Total: 12 bytes

**Message 16:**

|  |
| --- |
| User Name |
| Variable length string |
| 4 ~ 36 bytes |

Total: 4 ~ 36 bytes

**Task 5:**

Assumptions:

1. Assume that the Catalog Server reply a response to Registration Client. If Registration Client can create/update a catalog in Catalog Server, the Catalog Server will reply “Y”, if not, reply “N”. So the length of reply is 1, and size is 4 bytes.
2. Assume that the User Server reply a response to Registration Client. If Registration Client can create/update a user in User Server, the User Server will reply “Y”, if not, reply “N”. So the length of reply is 1, and size is 4 bytes.
3. Assume that the maximum number of authors is 7.
4. Assume that if this book is on hold to a different user, the Holds is “Y”, if not, the Holds is “N”. So the length of Holds is 1, and size is 4 bytes.
5. Assume that if flag is loan, then reply “L”, if flag is renewed loan, then reply “R”, if flag is hold, then reply “H”. So the length of flag is 1, and size is 4 bytes.

Limitation

1. If the number of book authors is more than 7, so the message cannot store the author which is after 7.
2. If the Loans Server is break, the CheckIn and CheckOut Client cannot use until the Loans Server is recover.
3. If the User Server is Overloading, it may makes Registration Client or Loans Server time-out.