

COMP6330 Assessment 3: Receiver

Here is the feedback and assessment mark for the Receiver assignment.

Please note the marks shown here are provisional as they haven't been checked by a second member of staff yet. Hence it may be a while before they appear on Kent Vision. You'll be notified if any changes are made.

OVERVIEW

Marks are divided into several categories, each corresponding to a group of related tests:

- 1) Frame decoding: short messages, no tricky chars.
- 2) Frame decoding: short messages, trickier content.
- 3) Frame decoding: short messages, with delimiter chars.
- 4) Message reassembly: longer messages, no tricky chars.
- 5) Message reassembly: longer messages, with delimiters.
- 6) MTU compliance: various message lengths and MTU values.
- 7) Robustness: frame format errors.
- 8) Robustness: segmentation/MTU errors & borderline cases.
- 9) Robustness: extreme errors & borderline cases.

For further details of how your work was marked see:

<https://www.cs.kent.ac.uk/~djb/comp6330/LOCAL-ONLY/general-feedback-receiver.html>

TEST RESULTS

Your implementation of `receiveMessage` was marked with the help of a partly automatic test rig. A summary of the results is below.

We regret it's not possible to reveal the specific frame strings, MTU values and other conditions used when testing your implementation. This is because we may wish to use a similar assignment in the future.

Each line in the results summary consists of several fields:

- test group number and part
- whether `receiveMessage` passed or failed
- how `receiveMessage` exited (see key at end)
- diagnostic codes (if any, see key at end)
- error message returned (e.g. if exception thrown)

TEST RESULTS SUMMARY

Test 1 : Frame decoding tests: short messages, no tricky characters

1a : Pass (Exit=N)
1b : Pass (Exit=N)
1c : Pass (Exit=N)
1d : Pass (Exit=N)
1e : Pass (Exit=N)
1f : Pass (Exit=N)

Test 2 : Frame decoding tests: short messages, trickier content

2a : Pass (Exit=N)
2b : Pass (Exit=N)
2c : Pass (Exit=N)
2d : Fail (Exit=E; Diags=ANE; ErrMsg="Error: MTU Is Not Large Enough Or Frame Length Doesn't")

Test 3 : Frame decoding tests: short messages containing delimiters

3a : Fail (Exit=N; Diags=MC,RWM,EFD)
3b : Fail (Exit=N; Diags=MC,RSM,EFD)
3c : Fail (Exit=N; Diags=MC,REM,EFD)
3d : Fail (Exit=N; Diags=MC,REM,EFD)
3e : Fail (Exit=N; Diags=MC,MCW,REM,EFD)
3f : Pass (Exit=N)

Test 4 : Message reassembly tests: longer messages, no tricky characters

4a : Pass (Exit=N)
4b : Pass (Exit=N)
4c : Pass (Exit=N)

Test 5 : Message reassembly tests: longer messages containing delimiters

5a : Fail (Exit=N; Diags=MC,RWM,EFD)
5b : Fail (Exit=N; Diags=MC,RWM,EFD)
5c : Fail (Exit=N; Diags=MC,RWM,EFD)
5d : Fail (Exit=N; Diags=MC,MCW,REM,EFD)
5e : Fail (Exit=N; Diags=MC,MCW,REM,EFD)

Test 6 : MTU compliance tests: various message lengths, various MTU values

6a : Pass (Exit=N)
6b : Pass (Exit=N)
6c : Pass (Exit=N)
6d : Pass (Exit=N)
6e : Pass (Exit=N)

Test 7 : Robustness tests: frame format errors

7a : Pass (Exit=E; ErrMsg="Error: Incorrect frame format (No bracket at beginning or end of
7b : Pass (Exit=E; ErrMsg="Error: Incorrect Hex Value.")
7c : Pass (Exit=E; ErrMsg="Error: Incorrect Hex Value.")
7d : Pass (Exit=E; ErrMsg="Error: Incorrect Hex Value.")
7e : Pass (Exit=E; ErrMsg="Error: Incorrect Hex Value.")

```

7f : Pass (Exit=E; ErrMsg="Error: Incorrect Hex Value.")
7g : Pass (Exit=E; ErrMsg="Error: MTU Is Not Large Enough Or Frame Length Doesn't Have Minin
7h : Pass (Exit=E; ErrMsg="Error: Incorrect Hex Value.")
7i : Pass (Exit=E; ErrMsg="Error: MTU Is Not Large Enough Or Frame Length Doesn't Have Minin
7j : Pass (Exit=E; ErrMsg="Error: Incorrect Hex Value.")
7k : Pass (Exit=E; ErrMsg="Error: Incorrect Hex Value.")
7l : Pass (Exit=E; ErrMsg="Error: Incorrect Hex Value.")
7m : Pass (Exit=E; ErrMsg="Error: Incorrect frame format (No bracket at beginning or end of
7n : Pass (Exit=E; ErrMsg="Error: Incorrect frame format (No bracket at beginning or end of
7o : Pass (Exit=E; ErrMsg="Error: Incorrect frame format (No bracket at beginning or end of
7p : Pass (Exit=E; ErrMsg="Error: Incorrect frame format (No bracket at beginning or end of

```

Test 8 : Robustness tests: segmentation/MTU errors & borderline cases

```

8a : Pass (Exit=N)
8b : Fail (Exit=E; Diags=ANE; ErrMsg="Error: MTU Is Not Large Enough Or Frame Length Doesn't
8c : Pass (Exit=N)
8d : Fail (Exit=N; Diags=MC,REM,EFD)
8e : Pass (Exit=N)
8f : Pass (Exit=N)
8g : Pass (Exit=N)
8h : Pass (Exit=E; ErrMsg="Error: No End (:) In Any Frames.")
8i : Pass (Exit=E; ErrMsg="Error: Incorrect Size For Frame.")
8j : Pass (Exit=E; ErrMsg="Error: MTU Is Not Large Enough Or Frame Length Doesn't Have Minin
8k : Pass (Exit=E; ErrMsg="Error: MTU Is Not Large Enough Or Frame Length Doesn't Have Minin
8l : Pass (Exit=E; ErrMsg="Error: Incorrect Size For Frame.")

```

Test 9 : Robustness tests: extreme errors & borderline cases

```

9a : Pass (Exit=N)
9b : Pass (Exit=N)
9c : Pass (Exit=N)
9d : Pass (Exit=N)
9e : Pass (Exit=N)
9f : Pass (Exit=N)
9g : Pass (Exit=N)
9h : Pass (Exit=N)
9i : Pass (Exit=N)
9j : Pass (Exit=E; ErrMsg="Forced ProtocolException thrown by receiveFrame")
9k : Pass (Exit=E; ErrMsg="Error: Incorrect Hex Value.")
9l : Pass (Exit=E; ErrMsg="Error: Incorrect Hex Value.")

```

Key to exit codes

N = Method under test exited normally

E = Method under test threw ProtocolException

Key to diagnostic codes

ANE = Abnormal exit but no error condition existed (i.e. test data was valid)

EFD = Spurious expanded field delimiter ";;" in message

MC = Message corrupt (error in content/format of message segment(s))

MCW = Many message characters are wrong/missing (more than 1/4)

REM = Spurious additional character(s) found after end of message

RSM = Spurious additional character(s) found before start of message

RWM = Spurious additional character(s) found embedded within message

Final mark: 78 out of 100