Course Conventions

1   Introduction

This document covers conventions that will be used throughout the course.

2   RFC Terms

For the project specifications in this class, we'll be using proper RFC terminology. It's the terminology you'll see if you ever implement protocols in the real world (e.g., IMAP or MCTCP), so it's good to get exposed to it now. In particular, we'll be using the keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" as defined in RFC 2119.

The terms we'll use the most in this class are "MUST", "MUST NOT", "SHOULD", "SHOULD NOT", and "MAY" (though we may use others occasionally), so we're including their definitions here for convenience (copied verbatim from the RFC):

- **MUST**   This word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification.

- **MUST NOT**   This word, or the phrase "SHALL NOT", mean that the definition is an absolute prohibition of the specification.

- **SHOULD**   This word, or the adjective "RECOMMENDED", mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.

- **SHOULD NOT**   This phrase, or the phrase "NOT RECOMMENDED", mean that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.

- **MAY**   This word, or the adjective "OPTIONAL", mean that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation which does not include a particular option MUST be prepared to interoperate with another implementation which does include the option, though perhaps with reduced functionality. In the same vein an implementation which does include a particular option MUST be prepared to interoperate with another implementation which does not include the option (except, of course, for the feature the option provides.)

3   Data Sizes

Data size specifications are notoriously arbitrary, contentious, and plentiful. Enough so that xkcd devoted a comic to it: https://xkcd.com/394/. In light of this fact, and of the fact that we use data size notation a lot in this course, we felt that it was worth it to explicitly define the conventions we'll be using throughout the course.

Sizes follow the following format:

```
<prefix><unit>
```

For example, kb means "kilobit" or 1000 bits, where "k" is the prefix (for "kilo"), and "b" is the unit (for "bit").

The unit can be either "b" for "bit" or "B" for "Byte".

The prefixes may be in one of two families: SI, or IEC. The SI prefixes are the standard ones you're used to from science class: "k", "M", and "G" for 1000, 1000^2, and 1000^3, and so forth (pronounced "kilo", "mega", and "giga" - and note that the "k" is lowercase, while the others are not). The IEC prefixes, on the other hand, are a bit more unusual. They are "Ki", "Mi", "Gi", and so on, using the same first letters as the SI prefixes. However, they don't correspond to powers of ten - they correspond to the powers two. Ki means 2^{10}, Mi means 2^{20}, and Gi means 2^{30}. They're pronounced just like their SI counterparts, except that the last two letters in the prefix are replaced with "bi". Thus, "kilo" becomes "kibi", "mega" becomes "mebi", and "giga" becomes "gibi".

To cement this a bit, here are a few examples: 1 kB (kilobyte) is 1000 bytes; 1 MB (megabyte) is 1000^2 bytes; 1 Kib (kibibit) is 2^{10} bits; 1 kb (kilobit) is 1000 bits; 1 Gb (gigabit) is 1000^3 bits.

Lastly, we will often talk not just about data sizes, but also data rates. For this, we simply add the "per unit" suffix. For example, for "kibibytes per second", we'd write "KiBps". For "kilobits per second", we'd write kbps.