Instructions on Plotting and Timing

Overview:[1]

Many of the programming assignments in CPSC 200 involve timing the execution of a function or program in order to compare the observed behaviour of the program against theoretical expectations. This means that the assignments are scientific experiments, and the gathering and display of the data should follow standard scientific practice. This handout briefly reminds you of what some of those practices are.

Properties of Graphs:

Every graph should have:

1. A title.
2. X and Y axes with the following properties:
   (a) the X axis should correspond to the independent variable, and the Y axis should correspond to the dependent (or measured) variable.
   (b) Each axis should be labelled and specify the units of measurement.
   (c) Each axis should have tick-marks and numbers making measurements easy to read.
3. Data points that are easy to identify. Where appropriate the data points should have error bars indicating known measurement inaccuracies. For instance, if timing data is known to be no more accurate than $\pm 10\,\text{ms}$ this should be indicated.

Multiplier	prefix	abbrev	Multiplier	prefix	abbrev
10^{-3}	milli	m	10^{3}	kilo	k
10^{-6}	micro	μ	10^{6}	mega	M
10^{-9}	nano	n	10^{9}	giga	G
10^{-12}	pico	p	10^{12}	tera	T

Figure 1: SI unit prefixes

Graphs should be quantitative, meaning that the reader should be able to determine values and make measurements from the graph. In this regard, commercial packages such as Excel should be used with caution, as these produce qualitative graphs more suitable for business presentations by default. It is often easier and faster to produce a quality graph using graph paper (sold in the bookstore) and a sharp soft pencil (HB or F).

[1]Minor revisions 2010-09-25. First prepared in 2006.

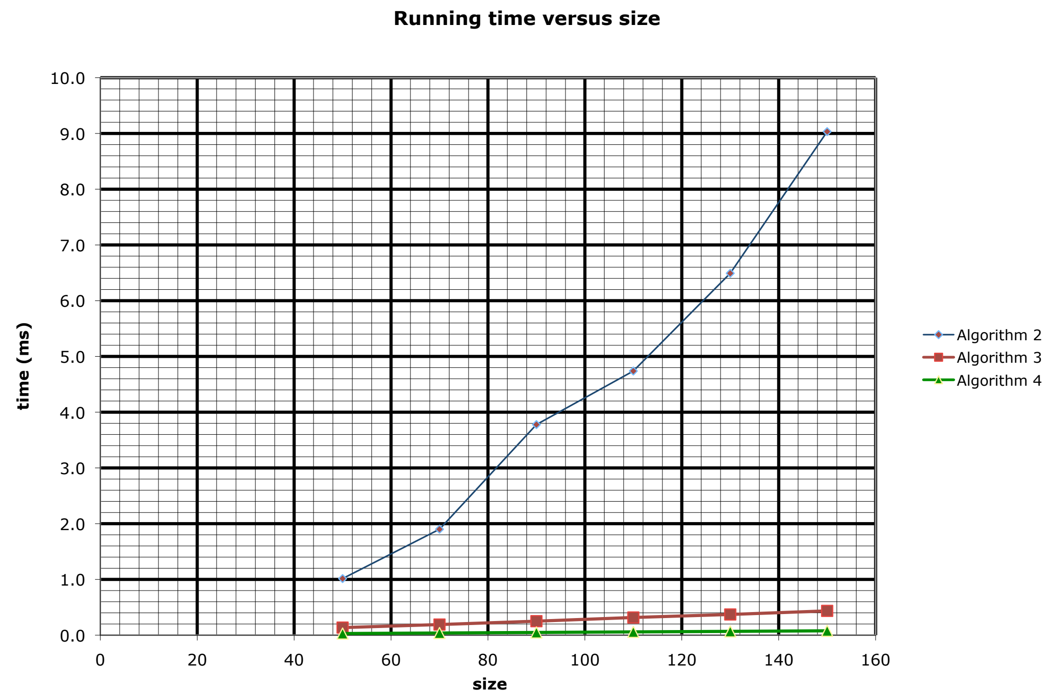

Figure 2: Sample plot

Choice of Units

Generally speaking, units should be chosen so that numbers are between 0.1 and 100.0. For instance, 20.1 ms is usually preferable to 0.0201 s. However, you should use the same units throughout one graph. Use standard SI abbreviations (See Figure 1).

Data Collection:

When plotting the time to run an algorithm versus the size of the problem, you need to write your program to produce data that will give you a good graph. What problem sizes should you choose? There is no fixed answer to this question, but in general you should try to find problems whose running times are somewhere between 1 s and 60 s. Running times that are shorter are less accurate, and running times that are much longer make waiting for the program to run painful. Similarly, the range of problem sizes to try will depend on the relationship between problem size and running time. If the relationship appears to be linear, or nearly linear, a large range (10-to-1) helps makes non-linearities easier to spot. If the relationship between problem size and running time is exponential, you may not be able to plot a 10-to-1 ratio in the independent variable. Try for approximately a 10-to-1 ratio in the dependent variable data. Produce data for evenly-spaced increments in the independent variable.

What to time:

Unless otherwise specified, you want to time only the execution of a particular algorithm and *not* the time to create the data, to return a result, or to print the data. Be very careful where you place your timing calls in your program to make sure that you don't accidentally time something that you didn't intend to.