frame=tb, language=Go, aboveskip=3mm, belowskip=3mm, showstringspaces=false, columns=flexible, basicstyle=, breaklines=true, breakatwhitespace=true, tabsize=3

Arrays and Performance

Willam Eriksson - werik@kth.se

Spring Fall 2022

Introduction

For the first task in problem set 1, we were tasked with benchmarking the access times on arrays with growing size.

Go for (int i = 0; i ¡ 100; i++) sum += i; for i := 0; i ¡ loops; i++ t0 := time.Now() sum += slice[r4nd] total += time.Now().Sub(t0) r4nd = rand.Intn(n)

This is what a report should look like, vanilla LaTeXwith regular page width and height and single spaced lines.

The name of the report should not be "My first report" and the name should not be "My Name". I thought that would be obvious but each year I have submitted reports were these templates have not been changed.

Layout

The rows in regular article mode are short - because it makes it easier to read. Do not set the column width or margins explicitly, let LaTeX decide what it should look like.

Don't use any fancy packages that will turn your report into a Christmas tree, keep it simple!.

sections

Since this is a small report I can omit having numbered sections and you do this by using section commands that end with a `*`. You can of course have subsections etc.

inserting code

Code snippets are included using the package `lstlisting`. If you want to include a program statement in running text you can do this using for example teletype-text: `List.sort()`.

java for (int i = 0; i ¡ 100; i++) sum += i;

The reports that you hand in should be four pages long - but not four pages of code! Use code snippets where you want to describe how things are done but don't include code just because you have written it.

numbers

You will include some run-time measurements in your reports. You should the think about the number of significant figures that you use. Just because a benchmark took $1.2345678s$ does not mean that you should report it in this way. If you write this in your report you're implicitly saying - if I do this again the number will be the same. This could be true but I doubt that anything you do on a computer can be determined with an 8 figure accuracy. The next time you try it might very well take $1.2354678s$. What you report is maybe $1.235s$ or $1.2s$?

tables

Numbers are often best presented in a table. You will have to do some reading on how to format tables but the general structures is quite easy. This is for example a table with some run-time figures.

prgm	**runtime**	**ratio**
dummy	115	1.0
union	535	4.6
tailr	420	3.6

Table 1: Union and friends, list of 50000 elements, runtime in microseconds

As you see in the table above, the run time per se might not be interesting. The interesting thing is how it relates to something else. Look at the ratios above, it gives you the information that we are looking for. So when you include numbers, ask your self why you have these numbers in the report. What is the purpose, can you describe it in a better way?

no f*ing screen shots

I know that you are all very happy that things actually work and eagerly want to show what things look like on you screen but please, don't use *screen shots*. It looks ugly and it's impossible to mark or copy the things that you want to show. It also, most often, show a lot of irrelevant things so instead of using an image, copy the text and format it so it's easy to read.

graphs

Once you start to generate graphs make sure that they are readable and have sensible information on the axes.

There are many ways to generate graphs but you want to use a way that minimize manual work. My tool over the years has been *Gnuplot* and if you do not have a favorite tool you could give it a try.

Gnuplot is not a statistical program nor a program that is very good at manipulating numbers but it is very good at taking a sequence of numbers and generate a nice graph.

LaTeXthings

Some LaTeXerrors that I frequently see that could easily be avoided if you only know where they come from.

less than

If you in your LaTeX code write "5 < 7" it will look like 5 ¡ 7 and "9 > 7" will look like 9 ¿ 7. Using the characters < and > directly does not work ... so, how did I do it? I used the commands `\textless` and `\textgreater` to generate the symbols < and >.

You could also use {`\tt 5 < 7`} but then it will use the teletype font and look like this: `5 < 7`.

Still another way is to write it using so called `math mode`. This is a mode used for writing mathematical formulas in a nice way. You enclose your expression in `$` signs like this `$5 < 7$` and then it will look like this $5 < 7$.

If you have a larger mathematical expression you enclose it in double $ and the result is that it is written centered with some space around it like this:

$$5 < (3 * 8/3)$$

why strange font

If you want to write *foo* in teletype font you write like this `{\tt foo}`. If you forget the closing } then it will look like this: `foo. Now everything here after until the end of you report will look like this.`