Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

Eliminating Small, Transient Memory Allocations

Dário Tavares Antunes
B. A. (Mod.) Computer Science
Final Year Project May 2018
Supervisor: Dr. David Abrahamson

School of Computer Science and Statistics
O'Reilly Institute, Trinity College, Dublin 2, Ireland

Declaration

I hereby declare that this project is entirely my own work and that it has not been submitted as an exercise for a degree at this or any other university.

_____

Dário Tavares Antunes, February 12, 2018

Permission To Lend

I agree that the Library and other agents of the College may lend or copy this report upon request.

_____

Dário Tavares Antunes, February 12, 2018

Abstract

This is where I would put the abstract, IF I HAD ONE

Acknowledgements

This is where I would put my acknowledgements, IF I HAD ANY

Contents

Chapter 1

Introduction

> 640K ought to be enough for anybody.
>
> ———————————
>
> Not Bill Gates

Despite the often misattributed epigraph above often being used to mock past beliefs that some amount of memory should be enough for any reasonable purposes, the mentality behind it is still pervasive.

With the broad availability of large amounts of computational power, memory and storage, conservation or efficient use of the same is often overlooked in programming. This is largely perpetuated by the (often valid) view that programmer time is more valuable than the benefits that more efficient but more complex code brings.

However, there remain situations where these benefits are in fact worth the effort required. One of these such cases is in code intended to be deployed in embedded or mobile devices, where resources are limited and preservation of power is essential.

A blog post [Ste17] by Daniel Stenberg, original author of the `curl` command line tool and ubiquitous URL data transfer tool, is a retrospective on an attempt to reduce unnecessary heap allocations.

Inspired by that post, the aim of this project is to produce a tool to identify cases where similar changes could be made in order to potentially reduce a program's energy and processing power footprint, and at the same time improve its performance.

Bibliography

[Ste17]   Daniel Stenberg. *Fewer mallocs in curl*. 2017. URL: https://daniel.
          haxx.se/blog/2017/04/22/fewer-mallocs-in-curl/ (visited on
          02/01/2018).