CAMBRIDGE

Cambridge University Press
978-0-521-81720-2 — Memory as a Programming Concept in C and C++
Frantisek Franek
Frontmatter
More Information

MEMORY AS A PROGRAMMING CONCEPT IN C AND C++

The overwhelming majority of program bugs and computer crashes stem from problems of memory access, allocation, or deallocation. Such memory-related errors are also notoriously difficult to debug. Yet the role that memory plays in C and C++ programming is a subject often overlooked in courses and books because it requires specialized knowledge of operating systems, compilers, and computer architecture in addition to a familiarity with the languages themselves. Most professional programmers learn about memory entirely through experience of the trouble it causes.

This book provides students and professional programmers with a concise yet comprehensive view of the role that memory plays in all aspects of programming and program behavior. Assuming only a basic familiarity with C or C++, the author describes the techniques, methods, and tools available to deal with the problems related to memory and its effective use.

Frantisek Franek is Professor of Computer Science at McMaster University, where he helped found the Algorithms Research Group. Franek's academic career encompasses research in mathematics (from the well-known Balcar–Franek theorem in Boolean algebra to finite combinatorics) as well as in computer science (from algorithms on strings to artificial intelligence). The author earned his Ph.D. at the University of Toronto and has held positions at several universities, including the Wesley Young Researchship at Dartmouth College. Franek has worked as a consultant on many commercial C/C++/Java projects internationally.

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-81720-2 — Memory as a Programming Concept in C and C++
Frantisek Franek
Frontmatter
More Information

MEMORY AS A PROGRAMMING CONCEPT IN C AND C++

FRANTISEK FRANEK
McMaster University

CAMBRIDGE
UNIVERSITY PRESS

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-81720-2 — Memory as a Programming Concept in C and C++
Frantisek Franek
Frontmatter
More Information

CAMBRIDGE
UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom

One Liberty Plaza, 20th Floor, New York, NY 10006, USA

477 Williamstown Road, Port Melbourne, VIC 3207, Australia

314-321, 3rd Floor, Plot 3, Splendor Forum, Jasola District Centre, New Delhi - 110025, India

79 Anson Road, #06-04/06, Singapore 079906

Cambridge University Press is part of the University of Cambridge.

It furthers the University's mission by disseminating knowledge in the pursuit of education, learning and research at the highest international levels of excellence.

www.cambridge.org
Information on this title: www.cambridge.org/9780521817202

© Frantisek Franek 2004

This publication is in copyright. Subject to statutory exception and to the provisions of relevant collective licensing agreements, no reproduction of any part may take place without the written permission of Cambridge University Press.

First published 2004
Reprinted 2005, 2006, 2007, 2009

A catalogue record for this publication is available from the British Library

Library of Congress Cataloging in Publication data
Franek, F. (Frantisek)
Memory as a programming concept in C and C++ / Frantisek Franek.
p. cm.
Includes bibliographical references and index.
ISBN 0-521-81720-X – ISBN 0-521-52043-6 (pb.)
1. Memory management (Computer science) 2. C (Computer program language) 3. C++ (Computer program language) I. Title.
QA76.9.M45F73 2003
005.4´35 – dc21 2003051543

ISBN 978-0-521-81720-2 Hardback
ISBN 978-0-521-52043-0 Paperback

Cambridge University Press has no responsibility for the persistence or accuracy of URLs for external or third-party internet websites referred to in this publication, and does not guarantee that any content on such websites is, or will remain, accurate or appropriate. Information regarding prices, travel timetables, and other factual information given in this work is correct at the time of first printing but Cambridge University Press does not guarantee the accuracy of such information thereafter.

CAMBRIDGE

Cambridge University Press
978-0-521-81720-2 — Memory as a Programming Concept in C and C++
Frantisek Franek
Frontmatter
More Information

CONTENTS

Acknowledgments
page ix

1
Introduction
page 1

2
From Source File to Executable File
page 7

Transformation of a source file to a load (executable) module. Why we can and do discuss source programs and their behavior as if they were executing somewhere in memory in their source form. Concepts of static memory allocation, dynamic memory allocation, program address space, and program system stack.

3
Variables and Objects; Pointers and Addresses
page 21

Variables as "data containers" with names. Values as data – simple (innate or elementary) data, structures, and objects. Referencing variables

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-81720-2 — Memory as a Programming Concept in C and C++
Frantisek Franek
Frontmatter
More Information

through pointers. Unnamed "data containers" and their referencing through pointers. The dual role of pointers as address holders and binary code "interpreters". Various interpretations of the contents of a piece of memory. Pointer arithmetic. Why C/C++ cannot be interpreted in a platform-free manner like Java can. Why C/C++ cannot have a garbage collector.

4
Dynamic Allocation and Deallocation of Memory
page 45

Fundamentals of dynamic allocation and deallocation of memory: free store (system heap); per-process memory manager; C memory allocators `malloc()`, `calloc()`, *and* `realloc()`; *and C deallocator* `free()`. *How to handle memory allocation/deallocation errors.*

5
Functions and Function Calls
page 59

System stack, activation frame, activation frame as the storage for local auto objects and for function arguments. Passing arguments by value as opposed to by reference. Calling sequence. Recursion and its relation to activation frames and the system stack. The price of recursion.

6
One-Dimensional Arrays and Strings
page 81

Static one-dimensional arrays and their representation as pointers. Array indexing as indirection. Why an array index range check cannot be performed in C/C++. The price of run-time array index range checking; the "compile-time checking" versus "run-time checking" philosophies. Passing static one-dimensional arrays as function arguments. Definition versus declaration of one-dimensional arrays. Dynamic one-dimensional arrays. Strings as static or dynamic one-dimensional `char` *arrays terminated with* `NULL`. *How to add a custom-made run-time index range checker in C++.*

7
Multi-Dimensional Arrays
page 97

Static multi-dimensional arrays and their representation. Row-major storage format and the access formula. Passing multi-dimensional arrays as function arguments. Dynamic multi-dimensional arrays.

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE
Cambridge University Press
978-0-521-81720-2 — Memory as a Programming Concept in C and C++
Frantisek Franek
Frontmatter
More Information

8
Classes and Objects
page 106

Basic ideas of object orientation; the concepts of classes and objects. Operators `new`, `new[]`, `delete`, *and* `delete[]`, *and related issues. Constructors and destructors.*

9
Linked Data Structures
page 132

Fundamentals, advantages, and disadvantages of linked data structures. Moving a linked data structure in memory, or to/from a disk, or transmitting it across a communication channel – techniques of compaction and serialization. Memory allocation from a specific arena.

10
Memory Leaks and Their Debugging
page 159

Classification of the causes of memory leaks. Tracing memory leaks in C programs using location reporting and allocation/deallocation information-gathering versions of the C allocators and deallocators. Tracing memory leaks in C++ programs: overloading the operators `new` *and* `delete` *and the problems it causes. Techniques for location tracing. Counting objects in C++. Smart pointers as a remedy for memory leaks caused by the undetermined ownership problem.*

11
Programs in Execution: Processes and Threads
page 187

Environment and environment variables, command-line arguments and command-line argument structure. A process and its main attributes – user space and process image. Spawning a new process (UNIX `fork()` *system call) from the memory point of view. Principles of interprocess communication; System V shared memory segments and "shared memory leaks". Threads and lightweight processes; advantages and disadvantages of threads over processes. The need to protect the "common" data in threads. Memory leaks caused by careless multithreading.*

A
Hanoi Towers Puzzle
page 210

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-81720-2 — Memory as a Programming Concept in C and C++
Frantisek Franek
Frontmatter
More Information

CONTENTS

B
Tracing Objects in C++
page 216

C
Tracing Objects and Memory in C++
page 227

D
Thread-Safe and Process-Safe Reporting and Logging Functions
page 234

Glossary
page 239

Index
page 255

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-81720-2 — Memory as a Programming Concept in C and C++
Frantisek Franek
Frontmatter
More Information

ACKNOWLEDGMENTS

Every book is to a significant degree a team effort; there are always many people essential for the book's publication, from the author(s) all the way to the editors and publisher. This book is no exception, and my sincere gratitude goes to all the people who contributed to its publication. My special thanks go to George Grosman, a musician and man of letters, for his help with the style and the grammar (English is not my mother tongue), and to Dr. Jan Holub, a postdoctoral Fellow in the Department of Computing and Software at McMaster University, for his careful reading of the manuscript and checking of the technical aspects of the text.

Please note that lengthier sections of code – as well as solutions to selected exercises – can be found on my website: `www.cas.mcmaster.ca/˜franek`.

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-81720-2 — Memory as a Programming Concept in C and C++
Frantisek Franek
Frontmatter
More Information

To my parents
Prof. Dr. Jiří and Zdeňka Franěk for everything;
and my mentors and best friends
Dr. B. Balcar DrSc., Czech Academy of Sciences,
Prof. Emeritus Dr. A. Rosa, McMaster University,
Honorable V. L. Rosicky, Consul of the Czech Republic,
formerly president of Terren Corp.
for everything I know about computers and mathematics;
and my wife Marie and children Jacob and Nicole,
for their love, support, and understanding

© in this web service Cambridge University Press www.cambridge.org