

Максютов Малик Сабитович

Искусство вычислений в эпоху параллельности: Осваиваем на C++.

М.: ЛЕНАНД, 2021. — 208 с.

В связи с интенсивным развитием средств «быстрого» создания приложений, не учитывающих развитие современных компьютерных архитектур, в наукоемких областях знаний возникает острая необходимость в высокоэффективных производительных вычислениях, учитывающих эти изменения. Предлагаемые на рынке «универсальные решатели», к сожалению, не всегда могут решить специфические проблемы в этих областях. Поэтому необходимость в написании эффективного вычислительного кода не только не теряет своей актуальности сегодня, но и требует от специалиста знаний в области современных моделей программирования. Эта книга сможет послужить помощником при разработке таких программ.

Работа будет полезна широкому кругу читателей — от студентов физико-математических специальностей вузов до специалистов в области высокопроизводительных вычислений.

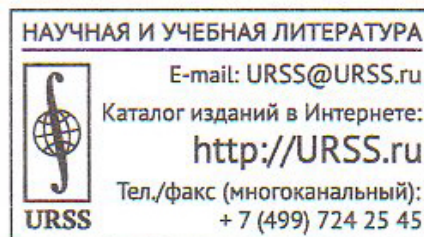
Формат 60×90/16. Печ. л. 13. Зак. № АР-8794.

Отпечатано в ООО «ЛЕНАНД».

117312, Москва, проспект 60-летия Октября, 11А, стр. 11.

ISBN 978–5–9710–8916–2

© ЛЕНАНД, 2021



Все права защищены. Никакая часть настоящей книги не может быть воспроизведена или передана в какой бы то ни было форме и какими бы то ни было средствами, будь то электронные или механические, включая фотокопирование и запись на магнитный носитель, а также размещение в Интернете, если на то нет письменного разрешения владельца.

Оглавление

Глава 1. Введение	5
Глава 2. Язык программирования C++	9
1. Конструкции языка	9
2. Структура программы.....	13
3. Типы используемых данных	15
4. Типы <i>указатель</i> и <i>массив</i>	22
5. Некоторые аспекты передачи параметров в функциях.....	31
6. Параметр <i>массив</i>	35
7. Обобщение символьных массивов. Тип данных <i>string</i>	44
8. Динамически выделяемая память	49
9. Дальнейшие расширения стандарта и некоторые параллельные конструкции при работе с массивами	55
10. Библиотека потокового ввода-вывода <i><iostream></i>	58
11. Операторы	63
12. Дополнительные операции ввода/вывода.....	70
13. Операции равенства, отношения и логические операции	78
14. Рекурсия и подстановка.....	81
15. Шаблоны	85

Глава 3. Основы объектно-ориентированного программирования	87
1. Структуры	87
2. Классы	93
3. Конструкторы	95
4. Деструкторы	99
5. Инкапсуляция, наследственность и полиморфизм	102
6. Основной класс Array и виртуальные функции	105
7. Производные классы	111
8. Классы C++ в работе	121
9. Шаблон класса	126
Глава 4. Оптимизация вычислений	131
1. Режимы отладки и оптимизации	131
2. Векторизация	142
3. Распараллеливание вычислений	151
Глава 5. Использование математических библиотек	159
1. Параллельное обращение матрицы. Проект MKL	161
2. Оценка эффективности распараллеливания	167
Приложения	174
Приложение 1. Векторно-матричные операции с классами	174
Приложение 2. Шаблон класса Array	194
Приложение 3. Работа с MKL	198
Список литературы	202

Глава 1

Введение

Читатель, держащий в руках эту книгу, невольно задаст вопрос: «Еще один „учебник“ по программированию»?! Спешу успокоить — это не совсем так. Более того, по большей части это не так. Да, необходимое здесь описание языка, его базового стандарта, безусловно, приводится. Но основное ее назначение в другом. Это не «книга о программировании», но «книга о вычислениях». Поэтому основной упор при описании языка сделан на данные и массивы. В XXI веке и сами вычисления претерпевают значительные трансформации, прежде всего в области параллелизма. Сегодня любое устройство, даже бытовое, содержит в себе многоядерный процессор. Задача его эффективного программирования становится настоящим искусством. Особенно важно использовать это искусство в таких чувствительных областях, как ядерная энергетика, космос и оборона.

Если вернуться к истории возникновения различных языков программирования, то можно обратить внимание, что изначальной областью их применения были вычисления. Вспомним хотя бы самый первый стандарт языка программирования, язык FORTRAN. До сегодняшнего дня ему нет равных по эффективности в научной и инженерной области деятельности. Да и сам язык претерпел значительные изменения, став фактически единственным нативным инструментом для параллельных вычислений. Тем не менее, за многие годы использования язык C++ тоже приобрел огромную популярность. Прежде всего, благодаря своей