

ВЛИЯНИЕ ПРОГРАММНОЙ РЕАЛИЗАЦИИ ВЫЧИСЛИТЕЛЬНОГО АЛГОРИТМА НА ЭФФЕКТИВНОСТЬ РЕШЕНИЯ ЗАДАЧИ

Л.В. Маркова, Н.Д. Адаменко, Е.В. Савостеенко
Витебск, ВГУ имени П.М. Машерова

Инновационное развитие общества сегодня напрямую связано с решением задач, требующих большого объема весьма сложных вычислений. Нет ни одной области науки и техники, в которой бы не ставились задачи математического моделирования. Это задачи ядерного синтеза, генной инженерии, тепло и массопереноса, экологических и метеопрогнозов, изучения человека, создания роботов и т.д. В основе вычислений каждой задачи лежат алгоритмы, важнейшим критерием разработки которых является их эффективность [1]. Цель исследования – показать возможность повышения эффективности решения задачи в зависимости от программной реализации вычислительного алгоритма.

Материал и методы. Объектом исследования была выбрана задача сортировки последовательных чисел. Основные методы исследования – системный подход, анализ литературы, методы ООП и параллельных вычислений, вычислительный эксперимент.

Результаты и их обсуждение. Очень часто при решении практических задач возникает необходимость расстановки последовательности чисел по возрастанию (неубыванию) и алгоритм сортировки используется в качестве промежуточного шага. Поэтому сортировка является основополагающей операцией в информатике и на ее примере можно демонстрировать многие стандартные методы разработки и анализа алгоритмов [1].

Исследования показали [2], что выбор языка программирования влияет на производительность труда программистов и качество создаваемого ими продукта. Для реализации поставленной цели выбранный язык программирования должен быть производительным, высокоуровневым и обеспечивать достаточно комфортные возможности реализации параллельного программирования. С учетом перечисленных факторов языком программирования был выбран C++.

Одним из важнейших критериев разработки алгоритмов является их эффективность, т.е. степень использования системных ресурсов компьютера. Эта характеристика учитывает такие факторы, как быстродействие программного продукта и требуемый для его исполнения объем памяти. В рамках проведенного исследования будем каждое описание алгоритма сопровождать анализом времени его работы.

Учитывая важность процедуры сортировки в решении практических задач, на сегодняшний день создано большое количество различных алгоритмов ее реализации. Для вычислительного эксперимента – сравнения эффективности процедуры упорядочивания элементов массива – были выбраны два вида алгоритма – пузырьковая сортировка и сортировка Шелла. Алгоритм пузырьковой сортировки – это сравнение и обмен двух соседних элементов исходной последовательности. Идея сортировки Шелла состоит в сравнении пар значений, располагающихся достаточно далеко друг от друга в последовательности, которую нужно упорядочить.

Вычислительный эксперимент проводился для последовательностей длиной от 250 до 60000 элементов. Результаты представлены на графиках (рис. 1).

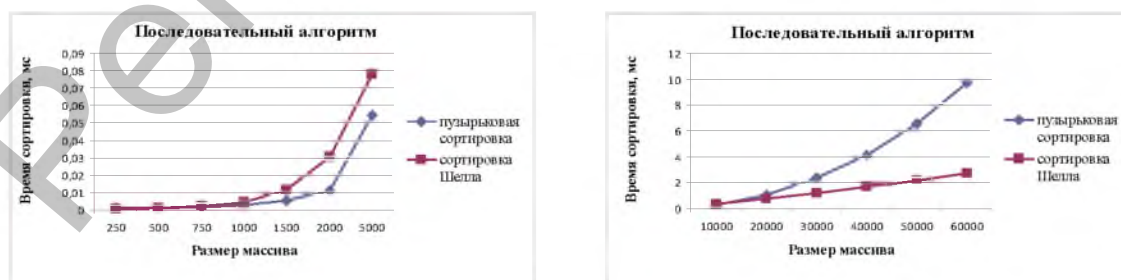


Рисунок 1

Пузырьковую сортировку принято считать низкоэффективной, однако эксперимент показал, что для последовательностей из 700–15000 элементов время ее выполнения гораздо меньше, чем у сортировки Шелла.

Производительность исполнения алгоритма можно увеличить путем уменьшения времени исполнения каждой операции микропроцессором. Однако это является очень сложной технической проблемой. Другим способом увеличения скорости вычислений является использование технологии параллельных вычислений, когда над решением одной задачи совместно работают несколько вычислительных устройств. При этом ставится цель – значительно уменьшить время решения задачи [3].

Алгоритм пузырьковой сортировки сложен для распараллеливания, поэтому для проведения исследований нами использовалась его модификация, состоящая в том, чтобы выполнять итерации в зависимости от четности или нечетности номера элемента последовательности. Сравнение же выделяемых элементов всегда осуществляется с их правым соседом. Для распараллеливания алгоритма сортировки Шелла использовалась аналогичная модификация метода. Анализ быстродействия проводился на процессоре Intel Core i5, включающий в себя 4 физических ядра. Результаты представлены на графиках (рис. 2).

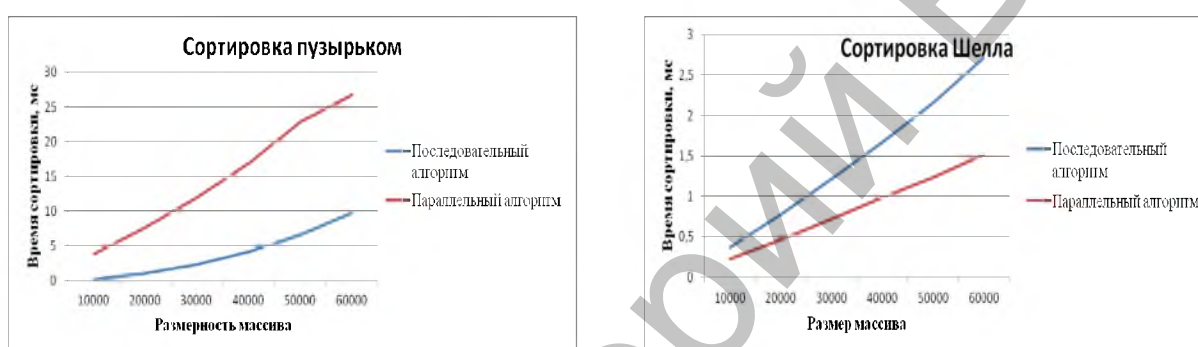


Рисунок 2

Заключение. Проведенные исследования показали зависимость эффективности решения задачи от вычислительного алгоритма. Выбор алгоритма должен в первую очередь определяться характеристиками поставленной задачи. В нашем случае такой характеристикой являлась размерность массива. Расчеты показали также, что увеличение числа процессоров не обязательно приводит к уменьшению времени решения задачи. Практическая значимость исследований состоит в том, что результаты могут быть использованы в учебном процессе для студентов IT-специальностей.

Список литературы

1. Кормен, Т. Алгоритмы: построение и анализ: пер. с англ. / Т. Кормен и др. – М.: ООО «И.Д. Ви льямс», 2013. – 1328 с.
2. Макконнелл, С. Совершенный код. Мастер-класс: пер. с англ. / С. Макконнелл. – СПб.: Издательско-торговый дом «Русская редакция», 2005. – 896 с.
3. Воеводин, В.В. Параллельные вычисления / В.В. Воеводин, Вл.В. Воеводин. – СПб.: БХВ-Петербург, 2002. – 608 с.

О РЕШЕТКЕ КОМПОЗИЦИОННЫХ ФОРМАЦИЙ С УСЛОВИЕМ ДОПОЛНЯЕМОСТИ

А.П. Мехович
Витебск, ВГУ имени П.М. Машерова

Все рассматриваемые группы конечны. Мы будем использовать стандартную терминологию из [1–3].

Напомним, что *формацией* называется класс групп, замкнутый относительно гомоморфных образов и конечных подпрямых произведений.

В произвольной группе G выберем систему подгрупп $\tau(G)$. Говорят, что τ – подгрупповой функтор (в терминологии А.Н. Скибы) [1], если выполняются следующие условия: