

УДК 32.973-018.2

## ИСПОЛЬЗОВАНИЕ СИСТЕМЫ РАСПРЕДЕЛЕНИЯ ПАМЯТИ НА ПРИМЕРЕ ВЫДЕЛЕНИЯ ПАМЯТИ ПОД МАССИВ ПСЕВДОСЛУЧАЙНЫХ ЧИСЕЛ СРЕДСТВАМИ ООП

*Пашкевич А.А.*

*УО «Барановичский государственный университет», г. Барановичи*

**ВВЕДЕНИЕ.** Алгоритмы отведения и распределения памяти – это особый интересный раздел теории и практики программирования [1]. На сегодняшний день мы имеем в своем распоряжении очень мощные компьютеры, которые за долю секунды обрабатывают огромное количество информации, начиная от простейших математических расчетов до симуляции сложных явлений [2]. Но какой бы мы производительный компьютер не имели, все утратит смысл без должной оптимизации программного обеспечения. Одним из способов увеличения быстродействия программы является внедрение в нее системы распределения динамической памяти.

Такая система позволит программам с использованием статических элементов либо динамических структур увеличить их эффективность и скорость обработки данных, при этом уменьшить количество выделяемой памяти и количество вызовов сложных системных механизмов распределения памяти [3].

Автором создана программа, которая наглядно демонстрирует различные алгоритмы распределения памяти, сравнивая их между собой по быстродействию и затраченной памяти на примере выделения памяти под массив псевдослучайных чисел. Для этого была разработана система распределения памяти блоками постоянной длины.

Объектом и предметом исследования является быстрое и эффективное выделение памяти под динамические структуры и ее освобождение, а также сравнение данной системы со статическим методом и стандартными функциями языка C++.

**ОСНОВНАЯ ЧАСТЬ.** Автором создана программа для наглядной работы алгоритмов распределения памяти:

- Создание входных данных происходит автоматически;
- Организация представления данных в таблице;
- Промежуточный вывод результатов для каждого из алгоритмов;
- Изменение размерности буфера и отображение его наполненности в системе распределения памяти;
- Вывод информации о системе, на которой запущено приложение;
- Графическое представление результатов в виде диаграмм.

Пользователю предлагается определить на начальном этапе входные данные. Для этого через меню Данные → Создать либо сочетанием клавиш Ctrl+N вызывается дополнительное окно «Создание данных», в котором можно задать тип данных (целочисленный или вещественный), размерность массива, диапазон данных и делитель для вещественного типа данных. После внесения необходимых параметров и нажатия кнопки «Создать данные» получаем готовые исходные данные (рисунок 1) и загружаем их на главное окно нажатием клавиши «Загрузить данные».

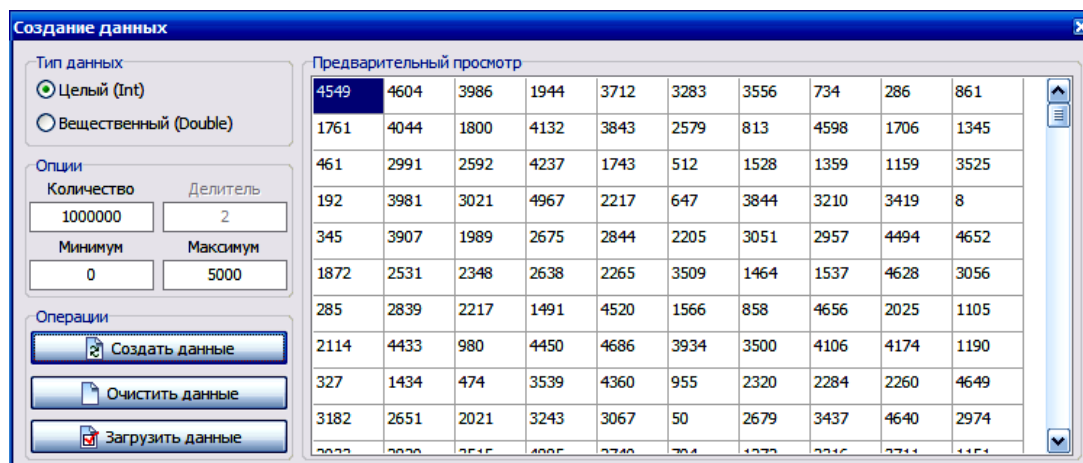


Рисунок 1 – Создание исходных данных

После создания исходных данных загружаем их в оперативную память любым из трех алгоритмов распределения памяти, получая промежуточные данные. Увеличивая размерность буфера системы распределения памяти, повышаем скорость выделения памяти под исходные данные (рисунок 2).

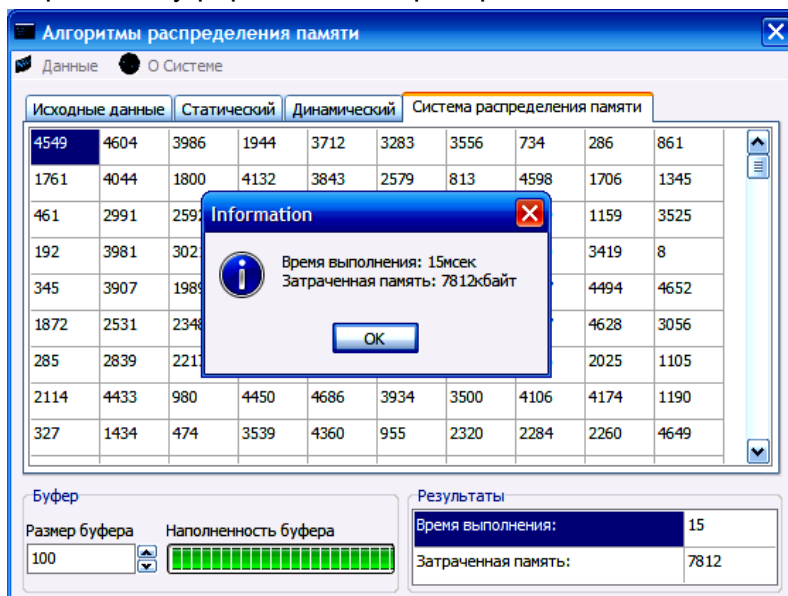
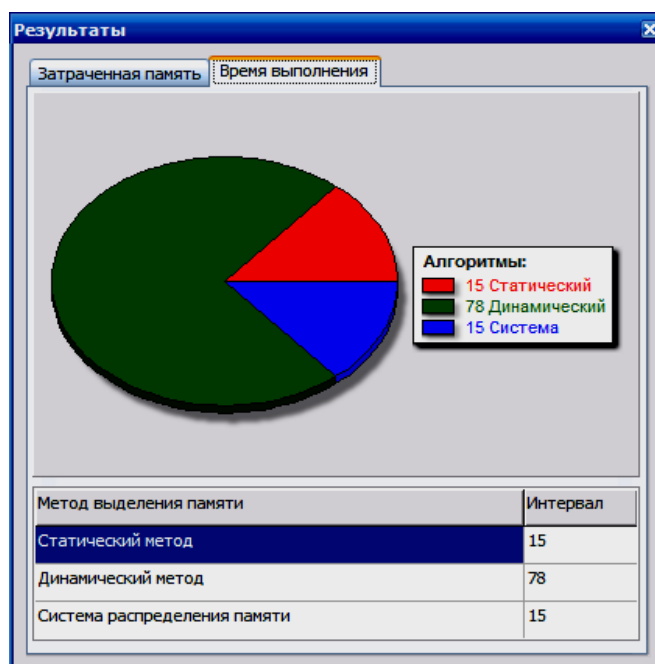


Рисунок 2 – Использование системы распределения памяти

Для наглядного представления выходных характеристик используются диаграммы. Программа демонстрирует пользователю время выполнения алгоритма и затраченную память на создание данных из исходных (рисунок 3).

Рисунок 3 – Результаты использования алгоритмов распределения памяти

Также для полной характеристики алгоритмов можно использовать информацию (касательно алгоритмов распределения памяти) о системе, на которых производилось исследование. Для этого достаточно открыть окно «О системе» из меню О системе (рисунок 4).



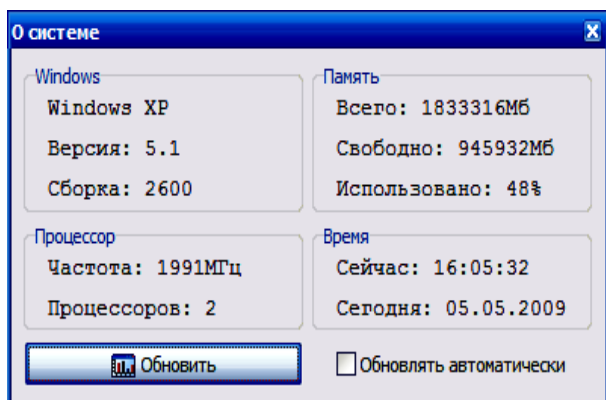


Рисунок 4 – Информация о системе

Данные исследования показали, что эффективно использовать систему распределения памяти возможно при размерах буфера, кратных размерности исходных данных, не малых, но и не слишком больших, и тем более не превышающих размерность исходных данных.

**ЗАКЛЮЧЕНИЕ.** В ходе исследования была разработана система распределения памяти блоками постоянной длины на примере выделения памяти под массив псевдослучайных чисел. Программа наглядно показывает работу алгоритмов распределения памяти, сравнивая их по быстродействию и затраченной памяти. Среда разработки CodeGear C++Builder 2009.

Системные требования:

- Дисковое пространство - 1.39Мб;
- Занимаемая оперативная память – от 5Кб (варьируется от размера массива и использованного для этого алгоритма);
- Операционная система –Windows 2000, XP, VISTA или 7 (необходима поддержка Unicode).

Данная работа рекомендуется также в качестве методического обеспечения при изучении раздела «Указатели» дисциплины «Конструирование программ и языка программирования» для студентов технических специальностей инженерного факультета БарГУ, может использоваться в системах с большим количеством входных данных, быстрым выделением памяти под них, например, в имитации ядерных реакций по распаду атомов урана, в видеомонтаже и обработки 3D-фигур.

Основные достоинства:

- Создание массива большой размерности и автоматическое его заполнение псевдослучайными числами с выбором типа и диапазона;
- Изменение размерности буфера;
- Информация о системе;
- Графическое представление результатов исследования.

К недостаткам можно отнести:

- Добавление/Удаление элементов из массива не реализована;
- Буфер имеет постоянную длину.

### Литература

1. Кубенский, А.А. Структуры и алгоритмы обработки данных: объектно-ориентированный подход и реализация на С++ / А.А. Кубенский – СПб.: БХВ-Петербург, 2004.
2. Вирт, Н. Алгоритмы и структуры данных / Н. Вирт; пер. с англ. – М.: Мир, 1989.
3. Романов, Е.Л. "Практикум по программированию на С++" / Е.Л. Романов – Санкт-Петербург, "БХВ-Петербург", 2004.