

Лабораторная работа №4  
**“Изучение алгоритмов распределения памяти.”**

**Цель**

Изучить алгоритмы распределения памяти и особенности их реализации.

**Варианты задания**

1. Реализовать менеджер памяти с выделением блоков различного размера.
2. Реализовать менеджер памяти с перемещающимися блоками.
3. Реализовать менеджер памяти со страничным разбиением.
4. Реализовать менеджер памяти с сегментным разбиением.
5. Реализовать менеджер памяти со странично-сегментным разбиением.
6. Реализовать менеджер памяти со страничным разбиением, использующий принципы временной локальности (см. раздел кэш-памяти) для загрузки виртуальных страниц в физическую память.
7. Реализовать менеджер памяти сегментным разбиением, использующий принципы временной локальности (см. раздел кэш-памяти) для загрузки виртуальных сегментов в физическую память.
8. Реализовать менеджер памяти со странично-сегментным разбиением, использующий принципы временной локальности (см. раздел кэш-памяти) для загрузки виртуальных страниц в физическую память.
9. Реализовать менеджер памяти со страничным разбиением, использующий принципы пространственной локальности (см. раздел кэш-памяти) для загрузки виртуальных страниц в физическую память.
10. Реализовать менеджер памяти сегментным разбиением, использующий принципы пространственной локальности (см. раздел кэш-памяти) для загрузки виртуальных сегментов в физическую память.
11. Реализовать менеджер памяти со странично-сегментным разбиением, использующий принципы пространственной локальности (см. раздел кэш-памяти) для загрузки виртуальных страниц в физическую память.

*Для выполнения заданий необходимо реализовать функции выделения и удаления блока памяти, чтения и запись информации из блока памяти, а так же функцию инициализации модели менеджера памяти. Функции должны обеспечивать чтение и запись по произвольному адресу произвольного количества байт (если позволяет размер выделенной для этих целей памяти).*

*В рамках решения задачи необходимо придумать и провести эксперименты для реализованных моделей памяти, которые позволяют выявить недостатки модели (подробнее о недостатках можно узнать из рекомендованной на лекциях литературы). По результатам экспериментов построить графики и выявить закон, по которому они растут/убывают. Провести анализ полученных показаний и объяснить почему был получен именно такой результат.*

*Реализовать юнит тесты для тестирования разрабатываемого приложения.*

### *Технические требования*

**Разработка должна выполняться на языке программирования C.**

### *Практические рекомендации*

В файле `memory.h` содержится описание интерфейса менеджера памяти

Порядок решаемых задач:

1. Разработать структуры, описывающие распределение памяти.
2. Реализовать алгоритм выделения памяти.
3. Реализовать алгоритм удаления блока памяти.
4. Реализовать алгоритм чтения и записи в память.
5. Протестировать модель менеджера памяти.

### *Что должно быть в отчете*

1. Краткое описание задачи.
2. Описание методов, которые были использованы для решения задачи.
3. Графики, демонстрирующие проведение нагрузочного тестирования реализованной модели. С комментариями и пояснением, в чем заключается смысл нагрузочного теста, и какие недостатки он выявляет.

### *Этапы защиты лабораторной*

1. Показать отчет и оформленные исходные тексты лабораторной работы.
2. Показать работу программы.
3. Ответить на контрольные вопросы.

### *Контрольные вопросы*

1. Чем ограничивается максимальный размер физической памяти, которую можно установить в компьютере определенной модели?
2. Чем ограничивается максимальный размер виртуального адресного пространства, доступного приложению?
3. В каких случаях транслятор создает объектный код программы не в виртуальных, а в физических адресах?
4. Что такое “свопинг”?
5. Как величина файла подкачки влияет на производительность системы?
6. Почему размер страницы выбирается равным степени двойки? Можно ли принять такое ограничение для сегмента?
7. На что влияет размер страницы? Каковы преимущества и недостатки большого размера страницы?
8. Почему загрузка и выгрузка данных из кэш-памяти производится блоками?
9. Как обеспечивается согласование данных в кэше с помощью методов обратной и сквозной записи?