

# Управление реальной памятью

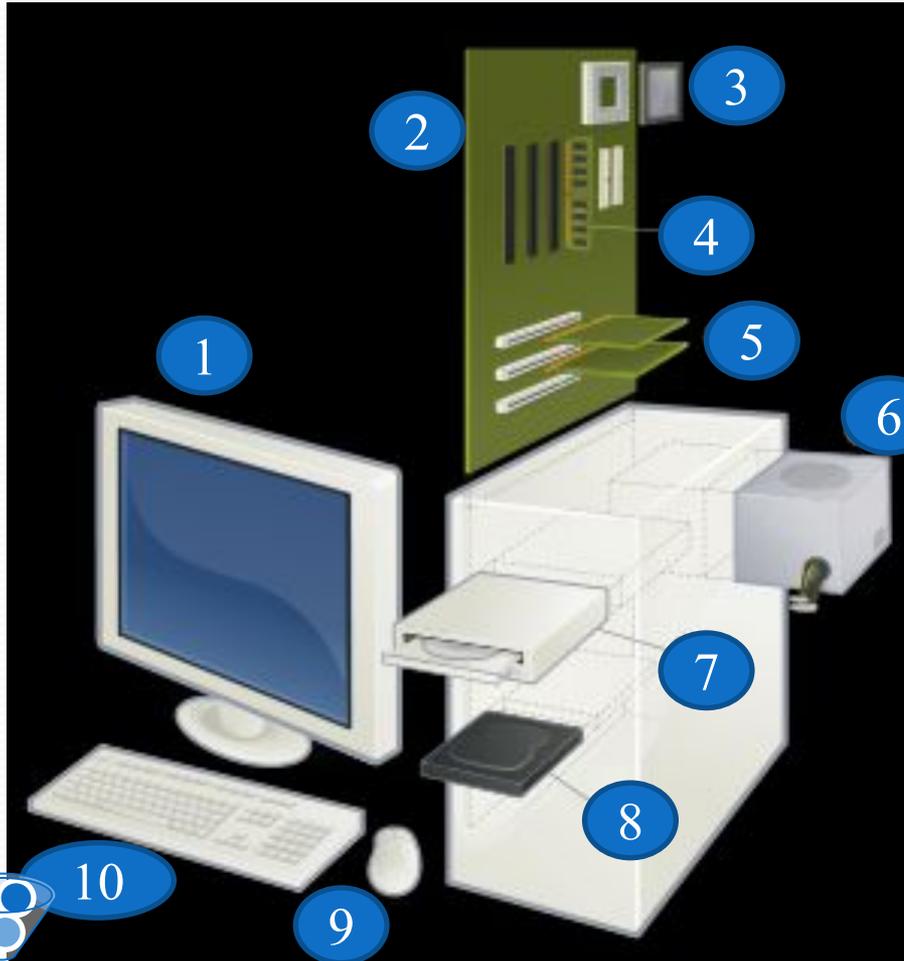
Механизм разделения центральной памяти.

Разделение памяти на разделы.

Распределение памяти с разделами фиксированного размера.

Распределение памяти с разделами переменного размера

# Основные составные части типичного персонального компьютера:



- 1 — Монитор,
- 2 — Материнская плата,
- 3 — Центральный процессор,
- 4 — Оперативная память,
- 5 — Карты расширений,
- 6 — Блок питания,
- 7 — Оптический привод,
- 8 — Жесткий диск,
- 9 — Компьютерная мышь,
- 10 — Клавиатура

**Перечислите  
и укажите на схеме  
известные вам  
разновидности памяти**

# Выберите правильные

## ОТВЕТЫ:

- Компьютерная память бывает:
  - Внутренняя постоянная память с произвольным доступом  
ПЗУ
  - Внешняя постоянная память с произвольным доступом  
НЖМД
  - Внутренняя оперативная память с произвольным доступом  
ОЗУ
  - Внешняя оперативная память с произвольным доступом

# Выберите правильные ОТВЕТЫ:

## ● К внутренней памяти компьютера относятся:

- 1) ПЗУ (ROM)
- 2) Flash BIOS
- 3) Flash-накопитель
- 4) CMOS-память
- 5) ОЗУ (RAM)
- 6) НЖМД

# правильные ответы:

- К внутренней памяти компьютера относятся:

- 1) ПЗУ (ROM)
- 2) Flash BIOS
- 3) Flash-накопитель
- 4) CMOS-память
- 5) ОЗУ (RAM)
- 6) НЖМД

# Выберите правильные

## ОТВЕТЫ:

- ЭнергоНЕзависимой компьютерной памятью является:
  - ОЗУ
  - ПЗУ
  - Flash BIOS
  - НЖМД
  - CMOS-память

# правильные ответы:

- ЭнергоНЕзависимой компьютерной памятью является:
  - ОЗУ
  - ПЗУ
  - Flash BIOS
  - НЖМД
  - CMOS-память

# Иерархия памяти

Обычное время доступа

Обычный объем



Поставьте соответствие: тип памяти и его

ра... жение



# Внутренняя память компьютера

## Внутренняя память компьютера

Внутренняя память компьютера состоит из двоичных запоминающих элементов — битов, объединенных в группы по 8 штук, которые называются байтами. Байты, в свою очередь, составляют ячейки памяти, называемые машинными словами; для разных моделей ЭВМ длина слова может быть различной, но количество входящих в него байтов всегда кратно двум. Память, ячейки которой состоят из указанного числа байтов, принято называть 8–, 16–, 32– и 64–разрядной.

**В состав внутренней памяти обычно входят:**

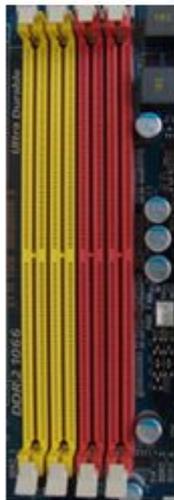
- Оперативная память**
- Кэш-память**
- Специальная память**

# Внутренняя память компьютера

## Оперативная память

Оперативная память (ОЗУ, англ. RAM — "Random Access Memory", "память с произвольным доступом") представляет собой быстродействующее запоминающее устройство сравнительно небольшого объема, непосредственно связанное с процессором и предназначенное для записи, считывания и хранения программ и обрабатываемых ими данных. Оперативная память может хранить в себе информацию, лишь пока компьютер работает, и полностью теряет свое содержимое при выключении питания, поэтому ОЗУ используется только для временного хранения программ и данных во время их выполнения/обработки. К ячейкам оперативной памяти обеспечивается прямой доступ, т.е. любая из них доступна процессору по ее индивидуальному адресу.

В современных компьютерах оперативная память обычно реализована на базе микросхем динамической памяти с произвольным доступом.



# Внутренняя память компьютера

## Кэш-память



Кэш-память (англ. «cache»), или сверхоперативная память — это очень быстрое ЗУ небольшого объема, используемое в качестве промежуточного "буфера" для хранения программ и данных в промежутке времени между их извлечением из ОЗУ и поступлением в процессор. Поскольку передача информации из ОЗУ в кэш и обратно производится независимо от работы процессора, это позволяет компенсировать разницу в скорости обработки информации процессором и обмена с несколько менее быстродействующей "обычной" оперативной памятью.

# Внутренняя память компьютера

## Специальная память

К устройствам специальной памяти относят постоянную память (ROM), перепрограммируемую постоянную память (Flash Memory), память CMOS RAM с автономным питанием от установленной на материнской плате аккумуляторной батарейки, видеопамять и некоторые другие виды памяти.



Постоянная память (ПЗУ, англ. ROM "Read Only Memory", "память только для чтения") — это энергонезависимая память (т.е. ее содержимое хранится в ней и при выключенном питании ЭВМ), используемая для хранения данных, не требующих изменения. Из ПЗУ информацию можно только читать.



Перепрограммируемая постоянная память (Flash Memory) — также энергонезависимая память, но допускающая многократную перезапись своего содержимого.

# Внешняя память компьютера



Основной функцией внешней памяти является **долговременное хранение большого объема информации.**

**Устройство, осуществляющее запись информации, называется накопителем информации;**

**Устройство хранения информации – носителем.**

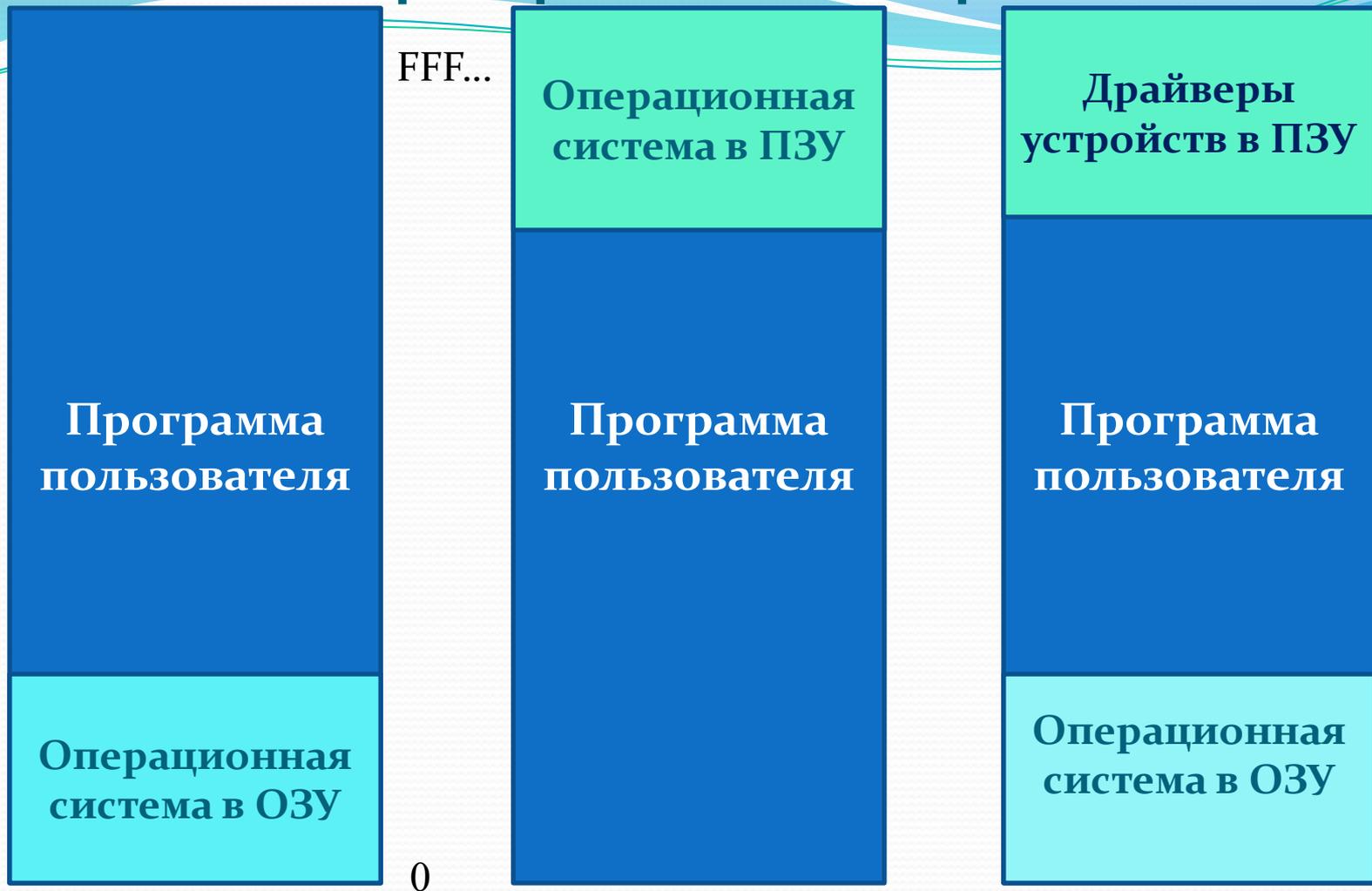
# Память – это ресурс, требующий четкого управления.

## Функции ОС по управлению памятью

### в однопрограммном режиме

- В ранних ОС управление памятью сводилось просто к загрузке программы и ее данных из некоторого внешнего накопителя (перфоленты, магнитной ленты или магнитного диска) в ОЗУ. При этом память разделялась между программой и ОС.

# В однопрограммном режиме



- **Первая** модель раньше применялась на мэйнфреймах и мини-компьютерах.
- **Вторая** схема **сейчас** используется на некоторых карманных компьютерах и встроенных системах.
- **Третья** модель была характерна для ранних ПК с MS-DOS.

# СВОПИНГ

- **Свопингом (перекачкой) называется метод управления памятью, основанный на том, что все процессы, участвующие в мультипрограммной обработке, хранятся во внешней памяти.**
- **Процесс, которому выделен ЦП, временно перемещается в основную память (swap in/roll in).**
- **В случае прерывания работы процесса он перемещается обратно во внешнюю память (swap out/roll out). Причем целиком, а не отдельной частью. Так как одновременно в памяти присутствует только одна программа конфликтов не возникает.**
- **Таким образом можно реализовать одновременный запуск нескольких программ.**

**□ Нарисуйте картинку, отображающую этот процесс.**

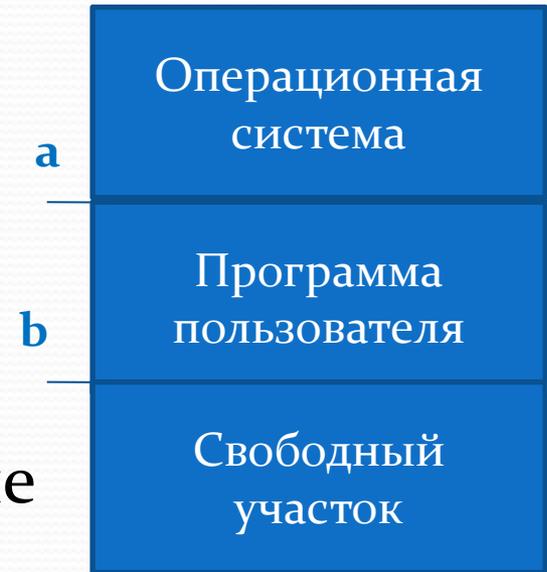
- Основное применение **СВОПИНГ** находит в системах разделения времени, где он используется одновременно со стратегией RR.
- В начале каждого временного кванта блок управления памяти выгружает из основной памяти процесс, работа которого была только что прервана, и загружает очередной выполняемый процесс.
- Метод свопинга влияет на величину временного кванта стратегии RR. Для эффективной загрузки ЦП время свопинга должно быть существенно меньше времени счета. Недостаток «чистого» свопинга заключается в больших потерях времени на загрузку и выгрузку процессов.
- Поэтому в современных ОС используются модифицированные варианты свопинга, например, свопинг включается только в том случае, когда количество процессов памяти становится слишком большим.

# Защита памяти

## В однопрограммном режиме

В процессе выполнения программы все ее адреса должны быть  $\geq a$ , иначе произойдет повреждение операционной системы.

Во время работы прикладной программы все адреса, генерируемые ЦП, сравниваются с содержимым регистра границы( $a$ ). Если генерируется адрес  $< a$ , работа программы прерывается.



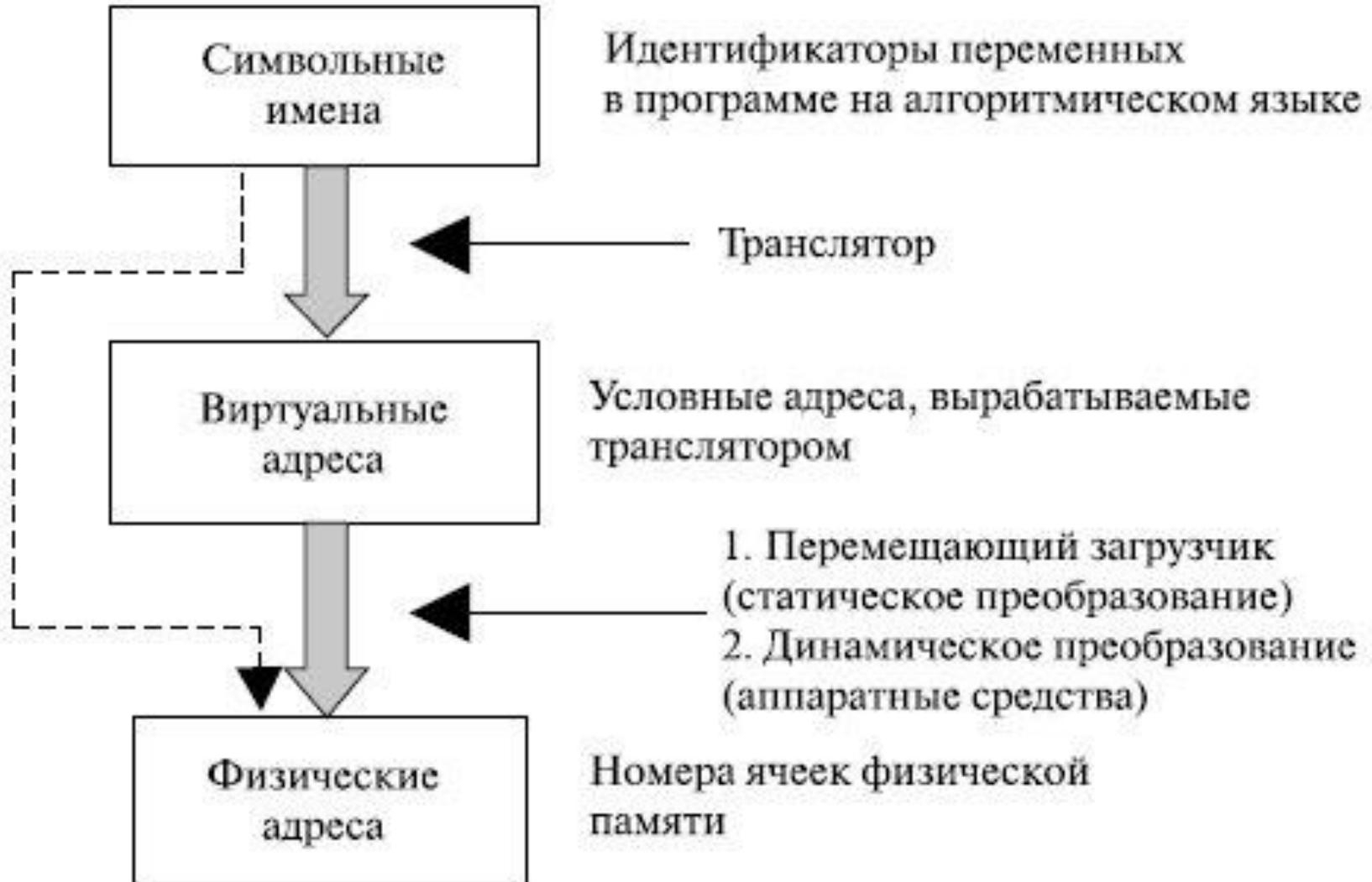
# Мультипрограммирование

е

**Функциями ОС по управлению памятью в мультипрограммных системах являются:**

- отслеживание (учет) свободной и занятой памяти;
- первоначальное и динамическое выделение памяти процессам приложений и самой операционной системе и освобождение памяти по завершении процессов;
- настройка адресов программы на конкретную область физической памяти;
- полное или частичное вытеснение кодов и данных процессов из ОП на диск, когда размеры ОП недостаточны для размещения всех процессов, и возвращение их в ОП;
- защита памяти, выделенной процессу, от возможных вмешательств со стороны других процессов;
- дефрагментация памяти.

Для идентификации переменных и команд на разных этапах жизненного цикла программы используются символные имена, виртуальные (математические, условные, логические – все это синонимы) и физические адреса



- Символьные имена присваивает пользователь при написании программ на алгоритмическом языке или ассемблере. Виртуальные адреса вырабатывает транслятор, переводящий программу на машинный язык. Поскольку во время трансляции неизвестно, в какое место оперативной памяти будет загружена программа, транслятор присваивает переменным и командам виртуальные (условные) адреса, считая по умолчанию, что начальным адресом программы будет нулевой адрес.
- Физические адреса соответствуют номерам ячеек оперативной памяти, где в действительности будут расположены переменные и команды.
- Существует два принципиально отличающихся подхода к преобразованию виртуальных адресов в физические. В первом случае такое преобразование выполняется один раз для каждого процесса во время начальной загрузки программы в память. Преобразование осуществляет перемещающий загрузчик на основании имеющихся у него данных о начальном адресе физической памяти, в которую предстоит загружать программу, а также информации, предоставляемой транслятором об адресно-зависимых элементах программы.
- Второй способ заключается в том, что программа загружается в память в виртуальных адресах. Во время выполнения программы при каждом обращении к памяти операционная система преобразует виртуальные адреса в физические.

# Методы распределения памяти

Без использования  
внешней памяти

Фиксированными  
разделами

Перемещаемыми  
разделами

Динамическими  
разделами

С использованием  
внешней памяти

Страничное  
распределение

Сегментное  
распределение

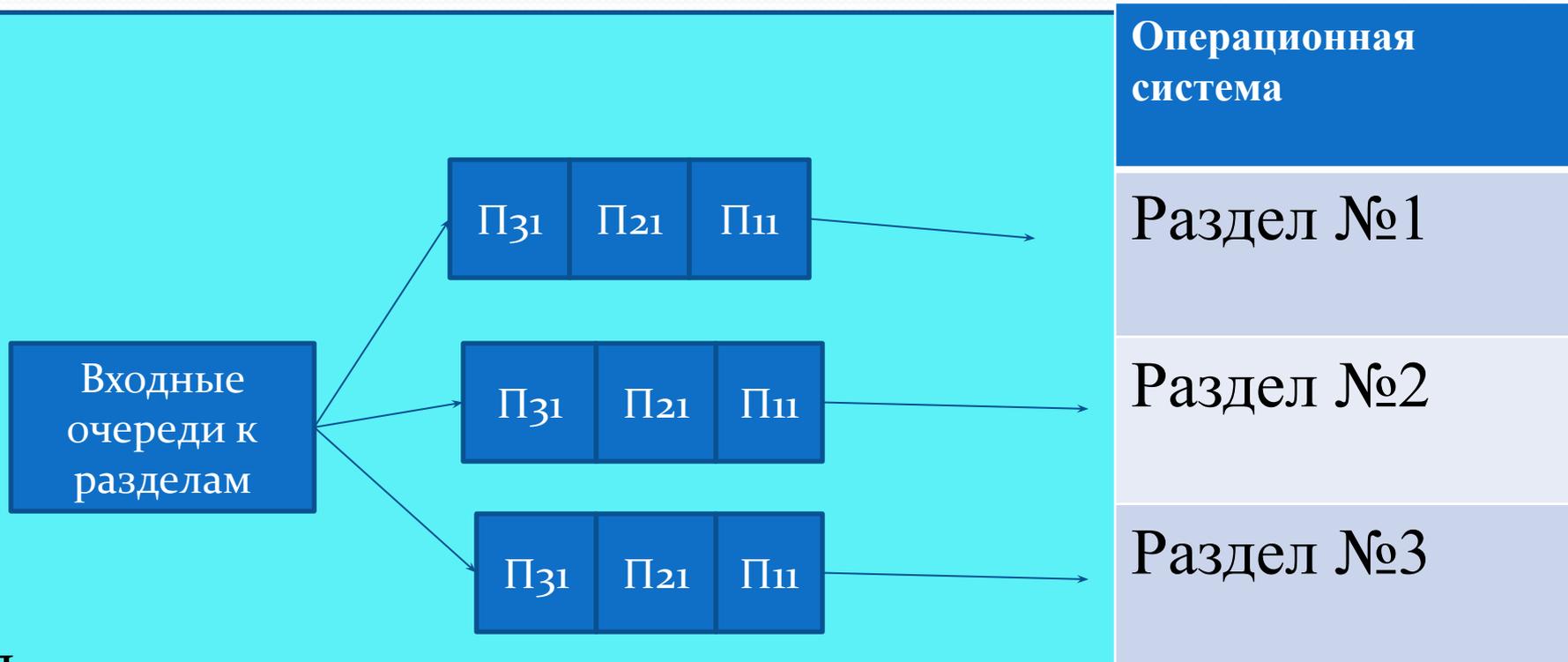
Сегментно-страничное  
распределение

# Методы размещения процессов

- Методы размещения процессов в основной памяти по отношению к расположению участков памяти, выделенной для одной и той же программы, делят на 2 класса:
  - Метод смежного размещения
  - Метод несмежного размещения
- **Смежное размещение** является простейшим и предполагает, что в памяти начиная с некоторого начального адреса, выделяется один непрерывный участок адресного пространства.
- **При несмежном размещении** программа разбивается на множество частей, которые располагаются в различных, необязательно смежных (расположенных рядом) участках адресного пространства.

# Мультипрограммирование с фиксированными разделами

(MFT-multiprogramming with a fixed number of tasks)



Предполагает разделение адресного пространства на ряд разделов фиксированного размера, в каждом разделе помещается один процесс.

Если соответствующий адресам процесса раздел занят, процесс остается в очереди во внешней памяти даже в том случае, когда другие разделы свободны.

# Мультипрограммирование с фиксированными разделами

- Для защиты памяти при MFT необходимы 2 регистра :
  - регистр верхней границы(наименьший адрес)
  - регистр нижней границы (наибольший адрес).
- Прежде чем программа в разделе N начнет выполняться, ее граничные адреса загружаются в соответствующие регистры.
- В процессе работы программы все формируемые ею адреса контролируются  $a < \text{Адр} < b$ .
- При выходе программы за отведенные ей границы работа программы прерывается.

# Мультипрограммирование с фиксированными разделами

- Уменьшить фрагментацию памяти при МФТ можно, если загрузочные модули создаются **в перемещаемых адресах**. Такой модуль может быть загружен в любой свободный раздел после соответствующей настройки.
- При мультипрограммировании с трансляцией в перемещаемых адресах имеются 2 причины фрагментации:
- Размер загруженного процесса меньше размера, занимаемого разделом (внутренняя фрагментация)
- Размер процесса в очереди больше размера свободного раздела, и этот раздел остается свободным (внешняя)

# Мультипрограммирование с переменными разделами

(MVT-multiprogramming with a variable number of tasks)

- предполагает разделение памяти на разделы и использование загрузочных модулей в перемещаемых адресах, однако **границы разделов не фиксируются.**
- **В начальной фазе отсутствует фрагментация**, связанная с тем, что размер очередного процесса меньше размера, занимаемого этим процессом раздела. На этой фазе причиной фрагментации является несоответствие размера очередного процесса и оставшегося участка памяти. По мере завершения работы программы освобождаются отдельные разделы. В том случае, когда освобождаются смежные разделы, границы между ними удаляются и разделы объединяются.
- За счет объединения и слияния смежных разделов образуются большие фрагменты, в которых можно разместить большие программы из очереди. Таким образом, на фазе повторного размещения действуют те же причины фрагментации, что и для метода MFT.

# Мультипрограммирование с переменными разделами и уплотнением памяти

- может создаться ситуация, когда в памяти образуется множество малых фрагментов, каждый из которых мал для процесса, а в сумме – превышает размер процесса.
- **Уплотнением памяти** называется перемещение всех занятых разделов по адресному пространству памяти таким образом, чтобы свободный фрагмент занимал одну связную область.

# Мультипрограммирование с переменными разделами и уплотнением памяти

- На практике реализация уплотнения памяти сопряжена с усложнением ОС и обладает следующими недостатками:
  - В тех случаях, когда мультипрограммная смесь неоднородна по отношению к размерам программ, возникает необходимость в частом уплотнении, что расходует ресурс процессорного времени и компенсирует экономию ресурса памяти
  - Во время уплотнения все прикладные программы переводятся в состояние ожидания, что приводит к невозможности выполнения программ в реальном времени.

# Основные стратегии заполнения свободного раздела

- В том случае, когда освобождается очередной раздел, ОС должна выбрать один из процессов.
- Алгоритм выбора может использовать одну из следующих стратегий:
  - Стратегия наиболее подходящего – выигрыш в памяти
  - Стратегия первого подходящего
  - Стратегия наименее подходящего – в этом случае остающийся фрагмент часто достаточен для размещения еще одного процесса