1.В вычислительной системе с двухуровневой страничной организацией памяти среднее время доступа процессора к одному данному составляет 185 нс. Частота попаданий в ассоциативную память при обращении к данным (hit ratio) составляет 75%. Оцените время доступа процессора к оперативной памяти, если время обращения к ассоциативной памяти равно 20 нс.

Решение: Пусть t0 - время доступа к оперативной памяти, t1 - время доступа к ассоциативной памяти, tm - среднее время доступа к данным, h - hit ratio (=совпадение/общее кол-во). Если адрес страницы памяти лежит в ассоциативном регистре, то время доступа к данному составит t0+t1. Если адрес не лежит в ассоциативной памяти, то время доступа будет 2t0+t0+t1 (дополнительно дважды сходим в таблицу страниц за адресом страницы и затем за данным). Таким образом, tm = h\*(t1+t0)+(1-h)\*(t1+3t0). Отсюда вычисляем: t0 = 110 нс.

2.

В вычислительной системе с трехуровневой страничной организацией памяти время доступа процессора к оперативной памяти составляет 120 нс. Среднее время доступа процессора к одному данному составляет 230 нс. Оцените, частоту попаданий в ассоциативную память при обращении к данным (hit ratio), если время обращения к ассоциативной памяти равно 20 нс.

Решение: Пусть t0 - время доступа к оперативной памяти, t1 - время доступа к ассоциативной памяти, tm - среднее время доступа к данным, h - hit ratio. Если адрес страницы памяти лежит в ассоциативном регистре, то время доступа к данному составит t0+t1. Если адрес не лежит в ассоциативной памяти, то время доступа будет 3t0+t0+t1 (дополнительно трижды сходим в таблицу страниц за адресом страницы и затем за данным). Таким образом, tm = h\*(t1+t0)+(1-h)\*(t1+4t0). Следовательно,

h = (t1+4t0–tm)/(3 t0). Отсюда вычисляем: h = 75%.

3.

В вычислительной системе FOOLs использован процессор с системой команд с одним операндом в оперативной памяти, ассоциативной памятью и отстуствующим кэшем данных. Операционная система использует страничную организацию памяти с подкачкой страниц по требованию. Время доступа процессора к оперативной памяти составляет 1 микросекунду, время доступа к ассоциативной памяти — 0.2 микросекунды. Среднее время замещения страницы — 10 миллисекунд. В случае промаха при обращении к ассоциативной памяти вероятность page fault’а составляет 1 процент. Считая, что thrashing фиксируется при падении производительности процесса более чем в 2 раза по сравнению с его работой на неограниченной оперативной памяти, оцените пороговое значение частоты попадания в ассоциативную память (hit ratio), при котором трэшинг еще не возникает. Временами выборки команды и выполнения команды при наличии данного в процессоре пренебречь.

Решение: Пусть t0 — время доступа к оперативной памяти, t1 — время доступа к ассоциативной памяти, tpf — среднее время замещения страницы, h — hit ratio, f — частота page fault’ов. Время выполнения команды в нашей задаче совпадает с временем доступа к данному. При неограниченной оперативной памяти среднее время доступа к одному данному есть h\*(t0+t1)+(1-h)\*(t0+t0+t1) = t0+t1+(1-h)\* t0.

При реальной работе среднее время доступа к одному данному определим как

h\*(t0+t1)+(1-h)\*(1-f)\*(t0+t0+t1)+(1-h)\*f\*(t0+t0+t0+t1+tpf) = t0+t1+(1-h)\*t0+(1-h)\*f\*(t0+tpf).

Трэшинг наступит при условии 2\*(t0+t1+(1-h)\*t0) ≤ t0+t1+(1-h)\*t0+(1-h)\*f\*(t0+tpf).

Отсюда h = 1-(t0+t1)/(f\*(t0+tpf)-t0). Вычисляем h ≤ 1-1.2/(10001\*0.01-1) ~ 0.98788.

Трэшинг не наступит при большей частоте.

4.

Пусть у нас есть диск, насчитывающий 90 цилиндров (от 0 до 89). Время перемещения головки между соседними цилиндрами составляет 2 мс. В текущий момент времени головка находится на 40-м цилиндре и двигается в сторону уменьшения номеров цилиндров; очередь запросов на выполнение операций ввода-вывода содержит обращения к цилиндрам 25, 7, 50. По ходу обработки существующих запросов в очередь поступают новые по следующей схеме:

Нарисуйте диаграммы, показывающие в каком порядке будет обрабатываться последовательность запросов:для каждого из алгоритмов: FCFS, LOOK, SSTF. Считать, что алгоритм LOOK по ходу движения может обрабатывать вновь появившиеся запросы. Для алгоритма SSTF считать, что принятое решение о выборе запроса для обработки не может быть изменено при появлении нового запроса. Вычислите полное время обработки последовательности запросов (временами передачи информации и смены направления движения головок пренебречь).

Решение: очередь запросов 25, 7, 50

FCFS:

40->25->7->50->70->32->1 время: 2\*((40-25)+(25-7)+(50-7)+(70-50)+(70-32)+(32-1)) = 330 мс

LOOK:

40->25->7->32->50->70->1 время: 2\*((40-7)+(70-7)+(70-1)) = 330 мс

SSTF:

40->50->70->25->32->7->1 время:2\*((70-40)+(70-25)+(32-25)+(32-1)) = 226 мс

Номер цилиндра

70

32

1

Время поступления запроса, начиная от текущего момента (мс)

18

110

156

5. В файловой системе FOOLix для хранения информации о дисковых блоках, выделенных файлу, используется метод индексных узлов. Дисковые блоки имеют размер 1 Кбайт, размер индексного узла — 128 байт. Размер адреса в дисковом адресном пространстве — 32 бита. Индексный узел содержит 10 прямых адресов блоков данных, один адрес блока косвенной адресации, один адрес блока двойной косвенной адресации, один адрес блока тройной косвенной адресации. Остальные байты индексного узла служат для хранения атрибутов файла. Размер индексного блока совпадает с размером блока данных. Сколько всего дискового пространства (в байтах) потребуется для хранения файла с размером: a) 1 байт; b) 262144 байта; c) 273408 байт?

Решение:

Файл размером 1 байт. Для хранения отводим 1 блок данных, информация о его адресе заносится в индексный узел. Итого требуется 1024 байта + 128 байт = 1152 байта.

Файл размером 262144 байта. Для размещения данных требуется 256 дисковых блока. Для 10 из них задействуем прямую адресацию, для оставшихся 246 блоков потребуется блок косвенной адресации. Итого необходимо 262144 байт + 1024 байта (индексный блок) + 128 байт = 263296 байт.

Файл размером 273408 байт. Для размещения данных требуется 267 дисковых блока. Для 10 из них задействуем прямую адресацию, для 256 блоков — косвенную адресацию (1 индексный блок), для оставшегося 1 блока — двойную косвенную адресацию (1+1 = 2 индексных блока). Итого имеем 273408 + 1024 + 2\*1024 + 128 = 276608 байт.

я поступления запроса, начиная от текущего момента (мс)

18

110

156