Introduction to Operating Systems and SQL for Data Science

Practice 4 – Memory

Memory Hierarchy

Disk	RAM	Cache	Registers
Large	Medium	Small	Very small
Slow	Medium	Fast	Very fast
Persistent	Non persistent	Non persistent	Non persistent
Cheap	Medium	Expensive	Expensive

Address space

- Each process has its own address space
- Base register – start of process's address space
- Limit register – size of process's address space
- Logical address – an address from process's address space
- Physical address- the actual address in memory

Memory block

- Memory is divided to fixed size units (blocks)
- A block can contain memory of **only one process**

RAM Blocks	Process Memory
9	A
8	A
7	
6	D
5	
4	B
3	B
2	B
1	
0	

Reichman University

Words

- Blocks contains many words
- Word is the smallest storage unit
- Each address in memory maps to a word in the memory.

RAM Blocks	RAM Addresses	Process Memory
9	144-159	
8	128-143	B
7	112-127	
6	96-111	C
5	80-95	C
4	64-79	C
3	48-63	C
2	32-47	A
1	15-31	A
0	0-15	A

Reichman University

Example #1:

- Given:
  - Process A needs150 byte memory.
  - Block size is 64 byte
- How much blocks does process A needs?

Example #1:

- Given:
  - Process A needs150 byte memory.
  - Block size is 64 byte
- How much blocks does process A needs?
- **Solution:**
  - $\text{RoundUp}(\frac{\mathbf{150}}{\mathbf{64}}) = \mathbf{3}$
  - Process that uses part of the block considered as using the whole block

Reichman University

Example #2:

- Given:
  - Process A needs16 byte memory.
  - Block size is 64 byte
  - Word size is 2 byte
- How many addresses are needed to describe A’s memory?

Example #2:

- Given:
  - Process A needs16 byte memory.
  - Block size is 64 byte
  - Word size is 2 byte
- How many addresses are needed to describe A's memory?
- **Solution:**
  - $(\frac{16}{2}) = 8$
  - Each address maps to a word

Reichman University

Example #3:

- Given:
  - 32 bits Address BUS.
  - RAM size is 32 GB
  - Word size is 4 byte
- What is the maximal memory size a process can have (in bytes)?

Example #3:

- Given:
  - 32 bits Address BUS.
  - RAM size is 32 GB
  - Word size is 4 byte
- What is the maximal memory size a process can have (in bytes)?
- **Solution**:
  - Address size is 32 bits so there are 2^{32} addresses
  - Each address maps to a word
  - $2^{32} \times 4Byte = 2^{34}Byte = \mathbf{16GB}$

Reichman University

Swapping

- The memory of a running process should be in the RAM
- Memory allocation is continuous
- If there isn't enough place in RAM for a process memory allocation, we can "free" some space in RAM by swap out other's process memory
- **Swap out:** move process's memory from RAM to the disk
- **Swap in:** move process's memory from disk to the RAM

Memory state management

- Bitmap
- Linked list

Bitmap

- Data structure represent for each block if its free (1) or not (0)
- Advantage: easy to implement
- Disadvantage: finding “hole” for a process can be relatively slow

BLOCK	Bit
9	1
8	1
7	0
6	1
5	0
4	1
3	1
2	1
1	0
0	0

RAM	Process Blocks
450MB	A
400MB	A
350MB	
300MB	D
250MB	
200MB	B
150MB	B
100MB	B
50MB	
0MB	

Reichman University

Linked list

- We'll hold a bi-directions linked list that describes the memory state (holes & processes)
- Advantage: takes less time to find a "hole" for a process
- **When new process is created:** we look for a hole in the list, if the hole found is too big
  we will split it to two nodes (one for the process and one for the left space)
- **When a process finishes:** we take the process's node and turn it to a "hole"
  if as a result of that action we create two consecutive holes, we'll merge them to one.

Linked list

- **Which process should we swap out?**
  1. A process with the needed memory size, we might need its memory back soon
  2. Least Recently Used – the process who wasn't active for the longest amount of time

Linked list

- **Which hole to pick?**
  1. **First-fit: first on the list that matches, ruins big holes**
  2. **Best-fit: smallest hole that matches, hard to find and creates small holes**
  3. **Quick-fit: we'll keep different holes list (by sizes) from each list we'll choose first fit**
  4. **Worst-fit: biggest hole, hard to find and ruins big holes**

Reichman University

Question #1

- Given a swapping system with the following memory state

1. Write the holes list if the system uses **first fit** match
   3-6,8-9,12-14,16-20
2. Write the holes list if the system uses **best fit** match
   8-9,12-14,3-6,16-20

Process	A						C			F					B					
Address × 1KB	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

Reichman University

Question #2 – first fit

- Given:
  - system with continues memory allocation and swapping by first-fit match and LRU swap out mechanism.
  - 500 MB RAM
  - **5 processes:**
    A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB

- **Show memory state through the following execution order:**
  A,B,C,D,B,C,E,B,A,D,C,E

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB												
400MB												
350MB												
300MB												
250MB												
200MB												
150MB												
100MB												
50MB												
0MB												

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB												
400MB												
350MB												
300MB												
250MB												
200MB	A											
150MB	A											
100MB	A											
50MB	A											
0MB	A											

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB												
400MB		B										
350MB		B										
300MB		B										
250MB		B										
200MB	A	A										
150MB	A	A										
100MB	A	A										
50MB	A	A										
0MB	A	A										

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB												
400MB		B	B									
350MB		B	B									
300MB		B	B									
250MB		B	B									
200MB	A	A										
150MB	A	A	C									
100MB	A	A	C									
50MB	A	A	C									
0MB	A	A	C									

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB												
400MB		B	B									
350MB		B	B									
300MB		B	B									
250MB		B	B	D								
200MB	A	A		D								
150MB	A	A	C	C								
100MB	A	A	C	C								
50MB	A	A	C	C								
0MB	A	A	C	C								

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB					B							
400MB		B	B		B							
350MB		B	B		B							
300MB		B	B		B							
250MB		B	B	D	D							
200MB	A	A		D	D							
150MB	A	A	C	C	C							
100MB	A	A	C	C	C							
50MB	A	A	C	C	C							
0MB	A	A	C	C	C							

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB					B	B						
400MB		B	B		B	B						
350MB		B	B		B	B						
300MB		B	B		B	B						
250MB		B	B	D	D	D						
200MB	A	A		D	D	D						
150MB	A	A	C	C	C	C						
100MB	A	A	C	C	C	C						
50MB	A	A	C	C	C	C						
0MB	A	A	C	C	C	C						

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB					B	B						
400MB		B	B		B	B						
350MB		B	B		B	B	E					
300MB		B	B		B	B	E					
250MB		B	B	D	D	D	E					
200MB	A	A		D	D	D	E					
150MB	A	A	C	C	C	C	E					
100MB	A	A	C	C	C	C	E					
50MB	A	A	C	C	C	C	E					
0MB	A	A	C	C	C	C	E					

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB					B	B						
400MB		B	B		B	B						
350MB		B	B		B	B	E					
300MB		B	B		B	B	E					
250MB		B	B	D	D	D	E					
200MB	A	A		D	D	D	E					
150MB	A	A	C	C	C	C	E	B				
100MB	A	A	C	C	C	C	E	B				
50MB	A	A	C	C	C	C	E	B				
0MB	A	A	C	C	C	C	E	B				

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB					B	B						
400MB		B	B		B	B			A			
350MB		B	B		B	B	E		A			
300MB		B	B		B	B	E		A			
250MB		B	B	D	D	D	E		A			
200MB	A	A		D	D	D	E		A			
150MB	A	A	C	C	C	C	E	B	B			
100MB	A	A	C	C	C	C	E	B	B			
50MB	A	A	C	C	C	C	E	B	B			
0MB	A	A	C	C	C	C	E	B	B			

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB					B	B						
400MB		B	B		B	B			A	A		
350MB		B	B		B	B	E		A	A		
300MB		B	B		B	B	E		A	A		
250MB		B	B	D	D	D	E		A	A		
200MB	A	A		D	D	D	E		A	A		
150MB	A	A	C	C	C	C	E	B	B			
100MB	A	A	C	C	C	C	E	B	B			
50MB	A	A	C	C	C	C	E	B	B	D		
0MB	A	A	C	C	C	C	E	B	B	D		

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB					B	B						
400MB		B	B		B	B			A	A		
350MB		B	B		B	B	E		A	A		
300MB		B	B		B	B	E		A	A		
250MB		B	B	D	D	D	E		A	A	C	
200MB	A	A		D	D	D	E		A	A	C	
150MB	A	A	C	C	C	C	E	B	B		C	
100MB	A	A	C	C	C	C	E	B	B		C	
50MB	A	A	C	C	C	C	E	B	B	D	D	
0MB	A	A	C	C	C	C	E	B	B	D	D	

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Solution

Process	A	B	C	D	B	C	E	B	A	D	C	E
450MB					B	B						
400MB		B	B		B	B			A	A		
350MB		B	B		B	B	E		A	A		E
300MB		B	B		B	B	E		A	A		E
250MB		B	B	D	D	D	E		A	A	C	E
200MB	A	A		D	D	D	E		A	A	C	E
150MB	A	A	C	C	C	C	E	B	B		C	E
100MB	A	A	C	C	C	C	E	B	B		C	E
50MB	A	A	C	C	C	C	E	B	B	D	D	E
0MB	A	A	C	C	C	C	E	B	B	D	D	E

- **A– 250MB B- 200MB C- 200MB D- 100MB E- 400MB**

Reichman University

Issues with Swapping

- Moving whole memory of a process from RAM to disk and vice versa
- Disk read and write actions are very slow – **bad for performance**
- RAM is fast but **small (process memory is limited to RAM size in this method)**
- Solution:
  - Virtual Memory (not in recitation scope)

Reichman University