

## *TÖL401G - Stýrikerfi*

### **1. Explain the difference between a logical and a physical address and the role of the memory management unit (MMU) in translating addresses.**

- Physical address: The actual address in memory.
- Logical address: The address that the process sees.
- MMU: Memory Management Unit. Translates logical addresses to physical addresses.

### **2. Explain memory organisation and address binding when loading processes and how they relate to different types of MMUs.**

- Memory organisation: Memory is divided into fixed-size blocks called frames.
- Address binding: The process is loaded into a frame. The logical address is then translated to a physical address.
- MMU: Memory Management Unit. Translates logical addresses to physical addresses.
- Types of MMUs:
  - Base and limit registers: The logical address is added to the base register to get the physical address. The limit register is used to check if the address is valid.
  - Relocation register: The logical address is added to the relocation register to get the physical address.
  - Dynamic relocation register: The logical address is added to the relocation register to get the physical address. The relocation register is updated after each instruction.

### **3. Explain Swapping.**

- Swapping: Moving a process from main memory to secondary memory and vice versa.
- Swapping is used when there is not enough memory to hold all processes.
- Swapping is used when a process is blocked and another process can use the memory.
- Swapping is used when a process is terminated.
- Swapping is used when a process is created.
- Today swapping is used less and less because it is slow and inefficient.

### **4. Apply first-, next-, best-, and worst-fit strategies for allocating memory contiguously.**

- **First-fit:** Allocate the first hole that is big enough. Search starts at the beginning of memory.
- **Next-fit:** Allocate the next hole that is big enough. Search starts where the last request was satisfied.
- **Best-fit:** Allocate the smallest hole that is big enough. Search the entire list, unless the list is ordered by size.
- **Worst-fit:** Allocate the largest hole. Search the entire list, unless the list is ordered by size.

### **5. Explain the distinction between internal and external fragmentation.**

- External fragmentation:

- Total memory space exists to satisfy a request, but it is not contiguous.
- Internal fragmentation:
  - Free memory within the memory allocated to a process, that can be used to satisfy requests of the process.

## 6. Understand Paged MMUs (PMMUs) with page tables and translation lookaside buffers for speeding up lookups.

- PMMU: Paged Memory Management Unit.
- Page table: A table that contains the mapping between logical and physical addresses.
- Translation lookaside buffer: A cache for the page table. It is used to speed up the translation of logical addresses to physical addresses.

## 7. Translate logical to physical addresses in a paging system using PMMU.

- Logical address space is divided into pages of fixed size.
- Pages are mapped onto frames in the physical memory using a page table.
- Pages/frames have same size that is a power of 2, i.e. page size =  $2^n$ .
- Logical address  $A_L$  (having  $m$  bits) generated by CPU is divided into:
  - $(m-n)$  bits) page number  $p$
  - $(n)$  bits) page offset  $d$
- By looking up the frame number  $f$  onto which page  $p$  is mapped according to the page table, the resulting physical address  $A_p$  ( $m$  bits) is:
  - $(m-n)$  bits) frame number  $f$
  - $(n)$  bits) page offset  $d$

## 8. Describe hierarchical paging/multi-level paging.

- Two-level page table:
  - Frame level of the first level page table entry points to a frame containing the second level page table for all the pages represented by the first level page table entry
  - First level page table entry is set to invalid to indicate that all the pages represented by the first level page table entry are invalid (in this case, no frame containing second level page table is needed)

## 9. Describe applications of paging: memory protection, circumvent external fragmentation, shared memory

- **Memory protection:**
  - Each page table entry has a valid-invalid bit. When this bit is set to valid, the associated page is in the process' logical address space and is thus a legal page. When the bit is set to invalid, the page is not in the process' logical address space.
- **Circumvent external fragmentation:**
  - Paging can be used to circumvent external fragmentation. External fragmentation occurs when there is enough total memory to satisfy a request, but the available memory is not contiguous. Paging solves this problem by dividing memory into fixed-size blocks called frames and dividing logical memory into blocks of the same size called pages. When a process is loaded into memory, its pages do not need to be contiguous.

- **Shared memory:**

- Shared memory can be implemented by having two processes share the same page table entry.  
This means that the two processes share the same physical memory.