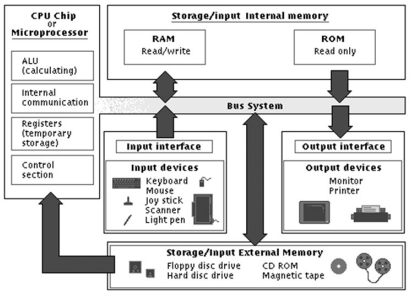
Audio visual - James Maclean [54038] 0161 306 4038

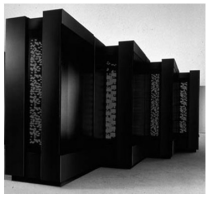


COMP25111
Operating Systems
Lectures 10
Memory Management (1)

Dr Richard Neville

r.neville@manchester.ac.uk

Room: G12 Kilburn Building, Bottom floor



Week

5

NOTE: The **up-to-date** version of this lecture is kept on the associated web site – available [on-line] @ Blackboard select: COMP25111 Introduction to Computer Systems **www.manchester.ac.uk/portal**

References:

1: Nova 840 (courtesy of Carl Friend) , available [on line] @ http://simh.trailing-edge.com/photos.html

2: Computer System, available [on line] @ http://encarta.msn.com/media_461532731/Computer_System.html

3: Connection Machine CM5, available [on line] @ http://hardware.localhost.nl/?ct=nth&showdir=pictures/2001-2003/Supercomputing

SkyDrive Real Time Video on

COMP25111 Lecture 11; "Intro To COMP25111 First Half" (Of the course), available on line @ WindowsLive_SkyDrive URL: https://skydrive.live.com/#cid=802FD99601DE4751&id=802FD99601DE4751!279

Learning; comprehension; & introspection

THIS week's Q1 Short Exam Questions

Have a go answering Q1 question below:

1. Question

   Name, the three components that make up a memory hierarchy as used in computer systems:

   ANSWER(S):

   The three components are:

   1. Processor;
   2. Primary memory [Main memory];
   3. Secondary [Hard disk].

- NOTE: In the exam approximately 2 question are taken from the topics (and program examples) coved in each lecture

THIS week's Short Exam Questions Q3

Have a go answering all Q3 questions below

1. Question

   Name, the three components that make up a memory hierarchy as used in computer systems:

   ANSWER(S):

   The three components are:

   1. Processor;
   2. Primary memory [Main memory];
   3. Secondary [Hard disk].

2. Question

   Which of the three is the fastest? Order them in terms of speed [access time].

   ANSWER(S):

   1. Processor approx. 10nS;
   2. Primary memory [Main memory] approx. 50nS;
   3. Secondary [Hard disk] approx. 5mS.

3. Question

   Order them in terms of size.

   ANSWER(S):

   1. Secondary [Hard disk] approx. 320GB.
   2. Primary memory [Main memory] approx. 2GB;
   3. Processor approx. 16 registers.

NOTE: In the exam approximately 2 question are taken from the topics (and program examples) coved in each lecture

Questions

Introduction to Questions:

The set of questions are based on lecture 11.

Answer Sheet will be given later in year and will contain the answers to these questions.

- Remember to find detailed and comprehensive answer you should [also] reference associated text books in the library.
- A reasonable starting place for associated book titles are:

1) This units 'module guide'; given to you in RN's first lecture – or on the web [Blackboard];
2) Those books mentioned in 'Background Reading;'
3) Those books [and web resources] mentioned in Learning Resources.

Questions

- **Questions (To Be Answered based on Lecture 11)**

1. Question

What steps are undertaken by an OS when it loads a uniprogram into memory?

Answer:

Questions

1. Answer to Question 1:

- OS undertake the following steps when a uniprogram is loaded into memory:
  1. OS given a file name;
  2. Loads it from disc;
  3. Jumps to start of program;
  4. Executes the program; may do I/O etc via OS;
  5. Returns to OS when done.

2. Question

What is the difference between:

a) absolute addresses;

b) physical Address; &

c) virtual address?

ANSWER(s):

2. Answer to Question 2:

- The difference are [in detail]:
  - **Absolute addresses**: An explicit identification of a memory location.
  - A fixed address in memory. The term absolute distinguishes it from a relative address, which indicates a location by specifying a distance from another location. Absolute addresses are also called real addresses and machine addresses.
  - **Physical Address**: un computing, a physical address, also real address, or binary address, is the memory address that is electronically (in the form of binary number) presented on the computer address bus circuitry in order to enable the data bus to access a particular storage cell of main memory.
  - **Virtual address**: Not real. The term virtual is popular among computer scientists and is used in a wide variety of situations. In general, it distinguishes something that is merely conceptual from something that has physical reality. For example, virtual memory refers to an *imaginary* set of locations, or addresses, where you can store data.
  - In computer terminology a virtual address is an address identifying a virtual (non-physical) entity. The term virtual address is most commonly used for virtual memory or virtual network address.

Questions

3. Question

What are base and limit registers? [In the context of relocation of programs in memory.]

Answer

"Base and limit registers are special hardware registers. When a process is run, the base register is loaded with the physical location where the process begins in memory. The limit register is loaded with the length of the process. In other words, they define the logical [allowable] address space [in the physical memory]."

They enable *virtual addresses* to be relocated to *physical address* space.

This is achieved by adding the base address (in the base address register) to the address from the CPU [virtual address] to calculate its actual physical address.

___

References:

1: Quote form: **Linux Open Source Enterprise Blog**, available [on-line] @ http://www.linuxvox.com/linux-notebook/memory-management/4-what-are-base-and-limit-registers

Questions

4. Question

In the case of a simple computer system with real memory. Explain why a multiprogrammed operating system would need to be able to relocate code? [In the context of relocation of programs in memory.]

Answer

In a multiprogrammed system, it is necessary to enable more than one program [to run] in the memory at one time. As it would be inefficient to save and load processes to and from disk every time a process switch occurred. In general it is not possible to know which combination of processes will be needed in memory; and it would be restrictive if a program had to be loaded into the same set of addresses every time [it was swapped in]. Hence, a scheme is necessary – where a program can be loaded starting at any convenient memory location, this is "relocation".

Questions

5. Question

Explain how a "loader program" can produce reloadable code while it is loading a program binary into main memory? [In the context of relocation of programs in memory.]

Answer

A program binary is a file which contains the machine instructions in a low level binary form[at]. If that program only uses relative branches then it is already reloadable and can be loaded directly. However, if the jump is an "absolute jump" then the target address will need to be modified. This is most easily done by storing the program with absolute addresses as though it were loaded at address 0 and then adding the actual start address to all targets [absolute jump addresses] as the program is loaded. These cam be identified by the loader if it knows the binary representation of absolute branches.

Questions

6. Question

With the aid of a diagram, describe the structure of a base and limit system and explain how it achieves relocation? [In the context of relocation of programs in memory.]

Answer

Draw a diagram like the one in your lecture notes – plus:

A base and limit system is a piece of hardware which is placed between the CPU and the memory. It has a 'base address' register which can be loaded as [and when] execution of a program is scheduled and this is added to any address issued by the processor (CPU). As the program is normally represented [formatted] in binary and nominally compiled to start a address 0, this will automatically address the program as if it started at the base address – held in the 'base address' register of the base+limits register.

The Limits register stops the program block [segment, or page] going beyond a maximum specified page size – nominally fixed in size…

Questions

7. Question

What other benefits does a base plus limit hardware provide? [In the context of relocation of programs in memory.]

Answer

The limits register is used to check that the address issued by the CPU is no greater than a value that can be loaded to a 'limit' register. By this we mean, the program can be prevented from accessing any memory outside a particular range and thus providing a degree of protection between multi [multiple] programs.

Questions

8. Question

What is 'swapping' in the context of operating system memory management and why is it useful? [In the context of relocation of programs in memory.]

Answer

In any multiprogrammed system, it is highly useful if any subset of a large number of processes can be resident in real memory at any one time. This means that a user can have the illusion that he/she is running a large number of processes, even if they won't fit into main memory all together. This is achieved by 'swapping'. Processes can be moved to and from [in and out of] background storage (disk) automatically by the system thus providing the above illusion.

Questions

9. Question

Explain briefly what is meant by the term multiprogramming?

Answer

Answer should cover the 'basic points' and for higher marks put these basic points into context.

Basic points:

Multiprogramming has been used in the past.

Multiprogramming loads multiple programs into the physical memory.

When using Multiprogramming the operating system switches between the multiple programs and switches them into the physical memory.

When one program has finished [the Multiprogramming operating system] bring in a new program.

Points contextualised:

In a multiprogrammed system, it is necessary to have more than one program in the memory at one time.

This is because it would be grossly inefficient to save and load processes to/from disk every time a process switch occurs.

As it is ,in general, impossible to know which combination of processes will be needed in memory, it would be very restrictive if a program had to be loaded at the same set of addresses every time.

Therefore, a scheme is needed where a program can be loaded starting at any convenient memory location, this is relocation.

Revision Exercises

- Scan read Lecture 11's Questions.
  - Answer Lecture 11's Questions
    - Particularly those questions you had difficulties with when you first tried them.