UG Exam Performance Feedback
Second Year
2016/2017 Semester 1

COMP25111 Operating Systems

Richard Neville
Oscar Palomar
William Toms

Comments

Will Toms:

A1a) Average mark: 0.75. In general this question was answered ok by most people who attempted it. A significant proportion of people got at either the size of the frame (0.2Mb) and the speed it takes over the wi-fi connection (5ms). Some people got confused between the speed of the connection and the frame-rate, and there were lots of mistakes with orders of magntitude in clock frequencies (millions=10^6, billions = 10^9). Interestingly the result was unaffected whether you chose to represent Mb as 10^6 or 2^20 (Mib), but some people also tried to use powers of 2 for the clock frequency which is always wrong.
A1b) Average mark: 0.9. There was quite a lot of confusion with this question with a lot of people thinking that it was a virtual memory question and stating that the address space contains a base and a limit. However, the actual answer, that the address-space contains all the addressable locations of a process was relatively easy for those who didn't get confused.
A1c) Average mark: 1.5. This answered correctly by most people. The main point was to differentiate between user mode and system mode and most people also identified that the reason for the two modes was to provide security.
A1d) Average mark: 0.8. The difference between the two types of operating system are that in a monolithic kernel all OS tasks are run entirely in system-mode, whereas in a micro-kernel only the bare minimum of protected operations are run in system mode and driver etc are run in user mode. A lot of people stated that monolithic kernels are "a big mess" which is not a technical term or particularly accurate - try telling that to Linus Torvalds!
A1e) Average mark: 1.1. This question was generally well answered. Some people thought that the race condition was some form a scheduling problem, but most people answered the question correctly.

A2a) Average mark: 1.4. This question was answered ok. The main point of the question was to highlight different forms of pre-emption, that were to be illustrated with a non-time slicing pre-emptive algorithm in the question A2d. Most people described pre-emption as strictly a time-slicing algorithm and so lost marks for additional examples of pre-emption. People also lost 1/2 mark for saying a process is pre-empted when it blocks for I/O - it is not.
A2b) Average mark: 1. This question was answered ok. The main point was to explain different types of static and dynamic priorities, such as user privilege or program defined (static) and cpu-boundedness or age (dynamic). More people got dynamic examples than static ones. A lot of peoples explanations seemed to be dictionary definitions of the words "static" and "dynamic" which is not what I was after.
A2c) Average mark: 1.9. This question was generally answered well. The main problems were simple mistakes, which are a problem with this type of question. Either people make a mistake transcribing numbers from the diagram or forget to put in a region for a process. One problem was that a previous years exam marking scheme gave a weird interpretation of waiting time (only the time between initially entering the ready queue and being dispatched to the run-queue), this does not match the lectures or any of the text-books and I described in the question exactly what metric I wanted, but some people still used this metric, it is unfortunate, but they only lost 1/2 mark.
A2d) Average mark: 1.5. This question was also answered ok. Again there were small errors in which people lost marks and some issues with the wrong wait-time metric. Some people didn't feature any pre-emption at all in there solutions which made it difficult to award any marks, while some other people who had made a real mess of A2c seemed to get this one right. The biggest problem, and one which I had to change the marking scheme retrospectively for, was that at time 13, both process A and process C had the same time remaining and were ready to run. In the question it stated that in this situation the process which had entered the ready queue first (A) was run first. I tried to maximise the marks of people who got everything right except making this mistake.

Oscar:

A2e) In general this question was answered well, but in a few cases there was some confusion with other file system concepts, like user and system level ids. Another mistake has been to attempt to (wrongly) link both ideas, for example saying that the FAT points to the i-nodes.

A2f)
Most people who attempted this question got the general idea of direct block pointers and indirection. However, some answers confused single indirection with double indirection, and double with triple, which resulted into much larger sizes. There were a few mistakes computing the size, or forgetting to multiply by the block size.

A2g)
Good answers in general by the people who attempted this question. One error I observed was to describe the algorithm for a file system structured with a “list of blocks”: this keeps the pointer to the next block in the data block itself, while a FAT-based file system keeps the indices separately in a dedicated table.

Another issue has been that some answers described the algorithm to parse the path to a file and find the first block of the file instead of describing the algorithm that uses that first block to access the whole file.

B3c)
Although the majority of answers were correct in describing the idea of deadlock, only a few were perfect, since many got some details wrong. Examples of problems were to restrict deadlocks to two processes only, or to indicate that all processes in the system must be involved.

Some answers failed to provide a good example of deadlock. There were simple examples with two threads and semaphores that were not correctly constructed, and did not result in a deadlock. The Dining Philosophers problem has been a popular choice; and while often it has been correctly described in a scenario in which it deadlocks, some answers assume this problem will always lead to a deadlock, while it may happen in certain cases and only if we use an implementation that is not deadlock-free.

B3d)
The most relevant type of error in the answers attempted was to make some type of timing assumption: i.e. that the scheduler will always select A, B, C in a specific order. This has become particularly apparent for part ii: wrong answers typically would work correctly if threads make progress following a specific sequence, but fail to enforce correct order of execution of the critical sections in case the scheduler makes a different decision.

Some answers show a correct sequencing of the semaphore functions and critical sections, but provide wrong final values for the semaphores or variables. I changed the marking to separate the semaphore and variable values.

Richard: please see the attached report.

General Feedback Comments

The following general comments are suggested to make your reflection and feedback more readable/succinct and viable for a more general succinct meditation on what you could do to enhance your learning and may be adapt your revision methodology.

First, it is important to reflect on the last lecture RN presented to the cohort; at the end of that lecture a comprehensive set of steps and guidance was presented for revision that students have advised me that they utilise. This was derived from methods students have utilised to revise over the years. The sentiments and guidance in this revision was sometimes given in their own words [the students]; and their own reflection on what worked best for different situations. But, may be it is worth noting that sometimes there can be discrepancies between the student's view of what mark they should have attained and what they actually were awarded. Reflecting on this issue may be it is worth noting and quoting the specific feedback from a student with respect to using the [suggested] self-test (or self-assessment) methodology, they said:

> "With regards to the self-assessment questions definitely allowed me to retain and recall large amounts of domain knowledge.
> It was especially useful in shorter questions and proved more beneficial than simply re-reading notes in a repetitive manner."

This is pertinent as without utilising a method like the self-assessment as well as undertaking a number of past exam papers one cannot self-access one's ability to pass the exam or access what mark one may obtain; and it is even more important when one goes into industry as without a good understanding of your own abilities how can you decide which courses to take, either those presented by the company that employs you or by external courses, and hence how can you evolve your personal skill set. You could also assesses where you feel you are in the **The Four stages of Learning (4SoL)): or Do you know what you know?**, information on this [4SoL] is on Blackboard 9.

Good companies will encourage you to undertake CPD. Continuing professional development (CPD) or Continuing professional education (CPE) is the means by which people maintain their knowledge and skills related to their professional lives.
One could say that evolving your revision and exam skill at University is, in fact, a form of CPD.

A final point, before getting into the detailed feedback for each question, is to reflect or ask yourself questions such as:
Did I undertaking past exam papers? [enough]
Did I undertaking a past exam paper –timed? [to get used to the time constraint of a real exam]
Did I develop a self-test (or self-assessment)? and finally
Did you utilise the methodologies presented in RN's exam revision lecture?

One of the recurring points was the advice on diagrams does not seem to be have been noted by many students; hence we repeat it hear: Remember, good – honours grade answer – in the exam – for a question – to maximise marks – should – or you should think of adding A DIAGRAM or a set of diagrams; hence a basic layout of a question answer may be:

1) Textual answer;
2) Diagram supporting answer [or code snippet]; &
3) Full explanation of diagram…

This sort of answer will [may] maximise your marks…
As per previous year's 75% of students answered section B2; which implies the majority (75%) of students answered questions from the second half of the lecture series. While, as per previous year's 80% of students answered section B3; which implies the majority (80%) of students answered questions from the first and second half of the lecture series.

Section B

B1. **Compulsory**

B1.a) In the context of converting an address generated by a program [a compiler] to the actual address; state:

i) The names of the two memories involved; &

ii) The unit that performs [undertakes] this translation process.

(2 marks)

B1.a)

Bookwork (2 marks):

a) In the context of converting an address generated by a program [a compiler] to the actual address; state:

i) The names of the two memories involved; &

ii) The unit that performs [undertakes] this translation process. (2 marks)

Example answer:- The following points should be covered to some degree in the answer:

In the context of converting an address generated by a program [a compiler] to the actual address:

i) The names of the two memories involved are:

a. Virtual memory; and

b. Physical memory.

ii) The unit that performs [undertakes] this process is undertaken by the MMU (Memory management unit).

2 marks for an answer that mentions all the salient facts in a sensible way (2 marks for a clear, correct answer, 1 mark for a 'right lines' solution),
1 ½ marks for correct answer but not detailed [enough],
1 mark for a right-lines approach,
½ marks for some basic understanding (or attempt).
Reference Learning Resources, Background Reading, and Lecture itself for detailed information; Lecture(s) No(s).: Lecture 10 & 11: Memory Management.
TOTAL marks (2 marks) [2]

Marker's feedback
Pedagogic assessment [criterion]:
The question assesses Lecture 10 & 11: Memory Management.

Well done, most of you were able to state what are the "names of the two memories involved; & the unit that performs [undertakes] this translation process in the context of converting an address generated by a program [a compiler] to the actual address".

The question's answer should clearly evidence knowledge of required salient facts relating to what are the "names of the two memories involved; & the unit that performs [undertakes] this translation process in the context of converting an address generated by a program [a compiler] to the actual address".

In the answer [some of] the following terminology (keywords and naming conventions) should be utilised in context; for example: Virtual memory, Physical memory, unit, performs, process, undertaken, MMU, Memory management unit; plus all those underlined in template answer in box above.

Main differentiation that must be clearly evidenced what are the "names of the two memories involved; & the unit that performs [undertakes] this translation process in the context of converting an address generated by a program [a compiler] to the actual address," are stated in the template answer.

The question's answer should clearly evidence knowledge of required salient facts relating to the issues aligned to what are the "names of the two memories involved; & the unit that performs [undertakes] this translation process in the context of converting an address generated by a program [a compiler] to the actual address"; this was not done explicitly in some of the answers given.

If the answers did not detail the mechanism plainly as stated in the above (and in the Example answer); e.g. if the answer does not give evidence of knowledge of required salient facts, full marks were not awarded.

This theory of what are the "names of the two memories involved; & the unit that performs [undertakes] this translation process in the context of converting an address generated by a program [a compiler] to the actual address" was covered in the lecture series in three different audio visual media: 1) the actual live lecture; 2) the audio recording of the live lecture; and 3) the real time video of the lecture.

B1.b) Differentiate between multiprogramming and fixed partitions. (2 marks)

B1.b)

Bookwork (2 marks):

Example answer:- The following points should be covered to some degree in the answer:

Multiprogramming:
Multiprogramming has been used in the past, to differentiate from an operating system (OS) running a single program [or Uniprogramming] and one that runs a number of programs concurrently (or multiprogramming).
The OS must first load the multiple programs (into memory [primary {physical} memory]).
The OS will then switch between them [the different programs]; this may be due to the program requiring I/O, or at regular intervals the OS will switch to another of the other programs.
When one of the programs is finished the OS brings in a new one.

Fixed partitions:
Fixed partition divides memory into fixed size blocks.
Fixed partitioning: involved partitioning the available primary memory into a number of regions with each region having a fixed size. The sum of the sizes of all regions [plus that used by the OS itself] equals the size of the primary memory.

2 marks for an answer that mentions the salient facts in a sensible way (2 marks for describing two of the above issues, one mark if one value identified correctly.),
1 ½ marks for correct answer but not detailed [enough],
1 mark for a right-lines approach,
½ marks for some basic understanding (or attempt).
Reference Learning Resources, Background Reading, and Lecture itself for detailed information; Lecture(s) No(s).: Lecture 10 & 11: Memory Management.
TOTAL marks (2 marks) [4]

Marker's feedback
Pedagogic assessment [criterion]:
The question assesses Lecture 10 & 11: Memory Management.

Well done, most of you were able to state what the "difference was between multiprogramming and fixed partitions".
The question's answer should clearly evidence knowledge of required salient facts relating to what the "difference was between multiprogramming and fixed partitions".
In the answer [some of] the following terminology (keywords and naming conventions) should be utilised in context; for example: single, Uniprogramming, number, programs, concurrently, multiprogramming, OS must first load, multiple, programs, memory, primary,

switch, between, the different programs, I/O, regular, intervals, other programs, finished, OS brings, new one [program],Fixed, partition, divides, memory, fixed, size, blocks, partitioning, available, primary, memory, a number, regions, fixed size; plus all those underlined in template answer in box above.

Main differentiation that must be clearly evidenced by what the "difference was between multiprogramming and fixed partitions," are stated in the template answer.

The question's answer should clearly evidence knowledge of required salient facts relating to the issues aligned to what the "differences were between multiprogramming and fixed partitions"; this was not done explicitly in some of the answers given.

If the answers did not detail the mechanism plainly as stated in the above (and in the Example answer); e.g. if the answer does not give evidence of knowledge of required salient facts, full marks were not awarded.

This theory of what the "difference was between multiprogramming and fixed partitions was covered in the lecture series in three different audio visual media: 1) the actual live lecture; 2) the audio recording of the live lecture; and 3) the real time video of the lecture.

B1.c) Differentiate between software interrupts and hardware interrupts; in your answer give a brief description of each type of interrupt. (2 marks)

B1.c)
Application, critique (2 marks),

Example answer:- The following points should be covered to some degree in the answer:

Basic description of two types of interrupt:

1. Software Interrupt (or Exception)

These occur when the processor meets a special interrupt instruction or when an error has occurred.

For example, executing a program that leads to an overflow in arithmetic calculation will lead to an overflow exception which will then cause some code to run – generally to inform the user of the overflow (a software interrupt is used).

2. Hardware Interrupt

The processor has a number of external connections called interrupt lines, these can be connected to external devices that can signal (by changing the logic level from, say, '0' to '1') that the external device wants to interrupt the processor.

2 marks for an answer that gives a concise description of both (2 marks for a correct answer, 1 mark for a 'right lines' approach. Moderate marks will be awarded in the case of correct application but calculation in error.)
1 ½ marks for correct answer but not detailed [enough],
1 mark for a right-lines approach,
½ marks for some basic understanding (or attempt).

Reference Learning Resources, Background Reading, and Lecture itself for detailed information; Lectures 14: Controlling Input and Output 1.

TOTAL marks (2 marks) [6]

Marker's feedback
Pedagogic assessment [criterion]:
The question assesses Lectures 14: Controlling Input and Output 1.

Well done, most of you were able to state what is the "difference between software interrupts and hardware interrupts; plus give a brief description of each type of interrupt".
The question's answer should clearly evidence knowledge of required salient facts relating to what the "difference between software interrupts and hardware interrupts; plus give a brief description of each type of interrupt.
In the answer [some of] the following terminology (keywords and naming conventions) should be utilised in context; for example: processor, interrupt, instruction, error, occurred, overflow, calculation, exception, processor, external, connections, interrupt lines, changing,

logic level, ‘0’ to ‘1’, external, device, wants to interrupt, the processor; plus all those underlined in template answer in box above.

Main differentiation that must be clearly evidenced by stating what is the “difference between software interrupts and hardware interrupts; plus give a brief description of each type of interrupt,” are stated in the template answer.
The question’s answer should clearly evidence knowledge of required salient facts relating to the issues aligned to what is the “difference between software interrupts and hardware interrupts; plus give a brief description of each type of interrupt”; this was not done explicitly in some of the answers given.
If the answers did not detail the mechanism plainly as stated in the above (and in the Example answer); e.g. if the answer does not give evidence of knowledge of required salient facts, full marks were not awarded.
This theory of what is the “difference between software interrupts and hardware interrupts; plus give a brief description of each type of interrupt was covered in the lecture series in three different audio visual media: 1) the actual live lecture; 2) the audio recording of the live lecture; and 3) the real time video of the lecture.

B1.d) State three distinct reasons why it is necessary for processes to share memory. (2 marks)

B1.d)
Bookwork and analysis (2 marks),

Example answer:- The following points should be covered to some degree in the answer:

In some cases it is necessary for processes to share memory:
1) Shared user code;
2) Shared data space (e.g. Unix pipes); or
3) Shared Library Code (dll's).

2 marks for an answer that mentions all the salient facts in a sensible way (2 marks for a reasonable coverage of the salient points, 1 mark for a 'right lines' answer),
1 ½ marks for correct answer but not detailed [enough],
1 mark for a right-lines approach,
½ marks for some basic understanding (or attempt).
Reference Learning Resources, Background Reading, and Lecture itself for detailed information; Lecture(s) No(s).: 13; Virtual Memory (3).
TOTAL marks (2 marks) [8]

Marker's feedback
Pedagogic assessment [criterion]:
The question assesses Lecture(s) No(s).: 13; Virtual Memory (3).

Well done, most of you were able to state "three distinct reasons why it is necessary for processes to share memory".
The question's answer should clearly evidence knowledge of required salient facts relating to what the "three distinct reasons why it is necessary for processes to share memory".
In the answer [some of] the following terminology (keywords and naming conventions) should be utilised in context; for example: shared; user code; data; space; library Code; dll's; plus all those underlined in template answer in box above.
Main differentiation that must be clearly evidenced by stating "three distinct reasons why it is necessary for processes to share memory;" are stated in the template answer.
The question's answer should clearly evidence knowledge of required salient facts relating to the issues aligned to stating "three distinct reasons why it is necessary for processes to share memory"; this was not done explicitly in some of the answers given.
If the answers did not detail the mechanism plainly as stated in the above (and in the Example answer); e.g. if the answer does not give evidence of knowledge of required salient facts, full marks were not awarded.
This theory of the "three distinct reasons why it is necessary for processes to share memory" was covered in the lecture series in three different audio visual media: 1) the actual live lecture; 2) the audio recording of the live lecture; and 3) the real time video of the lecture.

B1.e) In the context of permission information, e.g.: read (R), write (W), & execute (X), outline three different ways in which permissions are typically used? What can the RWX permission information control access to with respect to security and protection? (2 marks)

B1.e)
Bookwork (2 marks),

Example answer:- The following points should be covered to some degree in the answer:

Permission information: like R, W, & X. can be used to control access, for example to:
1) Code can be marked 'read only' (100);
2) Page tables can be marked 'read only' (100); or
3) Data can be marked 'read/write' but not 'execute' (110).

2 marks for an answer that mentions the salient facts in a sensible way (2 marks for correct answer, where answer is incorrect 1 mark may be awarded for use of a reasonable method if shown.);
1 ½ marks for correct answer but not detailed [enough],
1 mark for a right-lines approach;
½ marks for some basic understanding (or attempt).
Reference Learning Resources, Background Reading, and Lecture itself for detailed information; Lecture(s) No(s).: 13: Virtual Memory (3).
TOTAL marks (2 marks) [10]

Marker's feedback
Pedagogic assessment [criterion]:
The question assesses Lecture(s) No(s).: Lectures 13: Virtual Memory (3).

Well done, most of you were able to "outline three different ways in which permissions are typically used and what can the RWX permission information control access to with respect to security and protection".
The question's answer should clearly evidence knowledge of required salient facts relating to what the "three different ways in which permissions are typically used and what can the RWX permission information control access to with respect to security and protection".
In the answer [some of] the following terminology (keywords and naming conventions) should be utilised in context; for example: code; marked; read only; page tables; read only; data; read/write; not execute; plus all those underlined in template answer in box above.
Main differentiation that must be clearly evidenced by "outlining three different ways in which permissions are typically used and what can the RWX permission information control access to with respect to security and protection;" are stated in the template answer.

The question's answer should clearly evidence knowledge of required salient facts relating to the issues aligned to the "three different ways in which permissions are typically used and what can the RWX permission information control access to with respect to security and protection"; this was not done explicitly in some of the answers given.
If the answers did not detail the mechanism plainly as stated in the above (and in the Example answer); e.g. if the answer does not give evidence of knowledge of required salient facts, full marks were not awarded.
This theory of the three different ways in which permissions are typically used and what can the RWX permission information control access to with respect to security and protection was covered in the lecture series in three different audio visual media: 1) the actual live lecture; 2) the audio recording of the live lecture; and 3) the real time video of the lecture.

Section

B2.

B2.a) The following question address segmentation, which you were introduced to in your lecture series.

B2.a.1. In the context of memory management, what is a *segment*?

B2.a.2. What advantages are there in using segmentation?
Hint: What do segments *ensure* and *prevent*?

B2.a.3. Explain how logical addresses are translated into physical addresses in a system which has support for segmentation.

(6 marks)

B2.a)
Application (example re-ordering) (6 marks):

Example answer:- The following points should be covered to some degree in the answer:

B2.a.1. In the context of memory management, what is a *segment*?

"In a segmented virtual memory system, the segments are variable size block of memory."

Background Context: Segmentation is less about mapping a larger virtual address space onto a smaller physical memory (the purpose of paging), and more about **supporting** the computer system as a whole.

B2.a.2. What advantages are there in using segmentation? Hint: What do segments *ensure* and *prevent*?

Segments are a way of dividing [up] the virtual address space to support the management of the execution of multiple processes in an operating system:

1) ensure that processes do not interfere with one another;
2) ensure that operating system has control of the computer; and
3) will prevent programs written by users taking over the computer.

B2.a.3. Explain how logical addresses are translated into physical addresses in a system which has support for segmentation.
The logical addresses are divided into two fields:

b) Segment Number; &
c) Offset.

The Segmentation process of translation from logical (virtual) address to physical address:

The process steps are almost exactly the same as paged virtual memory; the steps are:
1) The processor generates a logical address;
2) The segment number field is used by the MMU (memory management unit) to look to see whether the segment is in memory or not, then:
3) If it is in memory, a physical address is computed by adding the base address of the segment to the offset; this is used as a physical address to memory;
4) If it is not in memory, the transfer is aborted (segment fault) and the operating system will load the segment from disk to memory.

6 marks for a concise description of complete segmentation loading process; names [correctly] both algorithms and concise descriptions,
3 marks for half issues covered,
1 mark for some basic understanding (or attempt).

Reference Learning Resources, Background Reading, and Lecture itself for detailed information; Lecture(s) No(s).: 12; Virtual Memory (2), Segmented Virtual Memory.

TOTAL marks (6 marks) [6]

Marker's feedback
Pedagogic assessment [criterion]:
The question assesses Lecture 12; Virtual Memory (2), Segmented Virtual Memory.

Well done, most of you were able to state what "is a *segment* in the context of memory management".
The question's answer should clearly evidence knowledge of required salient facts relating to what "is a *segment* in the context of memory management".
In the answer [some of] the following terminology (keywords and naming conventions) should be utilised in context; for example a.1, a.2. & a.3:
a.1: segmented; virtual; memory; segments; variable; size; block;
a.2: Segments; dividing [up]; virtual; address; space; support; management; execution; multiple; processes; operating system; ensure; processes; do not interfere; operating system; control; of the computer; prevent; programs; written; users; taking over; computer;
a.3: translation; from; logical; virtual; address; to; physical address; processor; generates; logical; address; segment number; MMU; memory management unit; in memory; or not, in memory; physical address; computed; adding; base address; segment; to the offset; used; physical; address; memory; not in memory; transfer; aborted; segment fault; operating system; load; segment; from disk to memory; plus all those underlined in template answer in box above.
Main differentiation that must be clearly evidenced when answering what "is a *segment* in the context of memory management;" are stated in the template answer.
The question's answer should clearly evidence knowledge of required salient facts relating to the issues aligned to clearly evidenced when answering what "is a *segment* in the context of memory management;" this was not done explicitly in some of the answers given.
If the answers did not detail the mechanism plainly as stated in the above (and in the Example answer); e.g. if the answer does not give evidence of knowledge of required salient facts, full marks were not awarded.
This theory of *segment* in the context of memory management was covered in the lecture series in three different audio visual media: 1) the actual live lecture; 2) the audio recording of the live lecture; and 3) the real time video of the lecture.

B2.b) In the context of page-replacement algorithms, explain the Not Recently Used (NRU) page replacement algorithm. (4 marks)

B2.b)
Bookwork and Critque (4 marks):

Example answer:- The following points should be covered to some degree in the answer:

The NRU algorithm makes sure recently used pages are kept in memory; it does this by replacing not recently used pages; it uses two parameters (bits) in this process; the R and M bits in a page table are used to:

1. The R bit set when referenced; and
2. The M bit set when modified (written to).

How it works:
At fixed intervals, the clock interrupt triggers and clears the referenced bit (R = 0) of all the pages.
Referenced bit marks pages referenced in interval.
So during interval if page referenced R=1 then it is used.
Else R=0 then it is NOT used;
and at the end of interval is a candidate for replacement.

4 marks for majority of above; concisely and explicitly covered,
3 marks for correct answer but not concise [enough],
2 marks for some information & for a right-lines approach,
1 mark for some basic understanding (or attempt).
Reference Learning Resources, Background Reading, and Lecture itself for detailed information; Lecture(s) No(s).: Lectures 13: Virtual Memory (3).

TOTAL marks (4 marks) [10]

Marker's feedback
Pedagogic assessment [criterion]:
The question assesses Lectures 13: Virtual Memory (3).

Well done, most of you were able to "explain the Not Recently Used (NRU) page replacement algorithm".
The question's answer should clearly evidence knowledge of required salient facts relating to "explaining the Not Recently Used (NRU) page replacement algorithm".
In the answer [some of] the following terminology (keywords and naming conventions) should be utilised in context; for example:
NRU; algorithm makes sure recently used pages; kept in memory; replacing; not recently used pages; uses two parameters; R; M; bits; page table; referenced; modified; fixed; intervals; clock; interrupt; triggers; clears; referenced bit; marks; pages; referenced in interval; R=0; NOT used; end of interval; candidate; replacement; plus all those underlined in template answer in box above.

Main differentiation that must be clearly evidenced when “explaining the Not Recently Used (NRU) page replacement algorithm,” are stated in the template answer.
The question’s answer should clearly evidence knowledge of required salient facts relating to the issues aligned to “explaining the Not Recently Used (NRU) page replacement algorithm”; this was not done explicitly in some of the answers given.
If the answers did not detail the mechanism plainly as stated in the above (and in the Example answer); e.g. if the answer does not give evidence of knowledge of required salient facts, full marks were not awarded.
To gain full marks candidates must explicitly state the conditions, in which, a page will be replaced in their answer; e.g. as the example template answer does (i.e. highlighted in yellow above), and you must also explicitly state which class of R & M, e.g. R=0 & M=0, are the pages to be replaced.
This theory explaining the Not Recently Used (NRU) page replacement algorithm was covered in the lecture series in three different audio visual media: 1) the actual live lecture; 2) the audio recording of the live lecture; and 3) the real time video of the lecture.

B2.c) Consider a process which accesses pages in the order 0,1,0,2,3,1,0,1,4,0,3,1.

Suppose that it has room in memory for exactly 3 pages, and that none of the pages mentioned above are loaded at the start of the sequence, what would be the number of page faults if Least Recently Used page-replacement algorithm is used? Draw up a 3 page frame memory table, as depicted in figure B2.c, to enable the calculation to be undertaken.

(5 marks)

Requested Order →

	0	1	0	2	3	1	0	1	4	0	3	1	
3 page frames													Most recent
													Second most
													Third most
PF													

M=Miss and H=Hit

Question B2.c. Diagram of a 3 page frame memory.

B2.c)
Application (5 marks),

Example answer:- The following points should be covered to some degree in the answer:

Request Order

	0	**1**	**0**	**2**	**3**	**1**	**0**	**1**	**4**	**0**	**3**	**1**	
3 page frames	0	1	0	2	3	1	0	1	4	0	3	1	Most recent
		0	1	0	2	3	1	0	1	4	0	3	Second most
				1	0	2	3	3	0	1	4	0	Third most
PF	M	M	H	M	M	M	M	H	M	H	M	M	

M=Miss and H=Hit
Total number of PF (page faults) after requested sequence was 9; using LRU page replacement.

5 marks for a totally correct content, 3 marks for all 3 page frames totally correct and 1 mark for the cumulative total number of page faults totally correct.
2 marks for half issues covered,
1 mark for some basic calculations (or attempt).

Reference Learning Resources, Background Reading, and Lecture itself for detailed information; Lecture(s) No(s).: 13; Virtual Memory (3).

TOTAL marks (5 marks) [15]

Marker's feedback
Pedagogic assessment [criterion]:
The question assesses Lecture 13; Virtual Memory (3).

Well done, most of you were able to "depict what would be the number of page faults if Least Recently Used page-replacement algorithm is used; and depict, in the diagram (in your answer), the order the pages are: loaded and rejected".
The question's answer should clearly evidence knowledge of required salient facts relating to "depicting what would be the number of page faults if Least Recently Used page-replacement algorithm is used; and depict, in the diagram (in your answer), the order the pages are: loaded and rejected".
In the answer [some of] the following terminology (keywords and naming conventions) should be utilised in context; for example: M; Miss; H; Hit; PF; page faults; 9; plus all those underlined in template answer in box above.
Main differentiation that must be clearly evidenced by "depicting what would be the number of page faults if Least Recently Used page-replacement algorithm is used; and depict, in the diagram (in your answer), the order the pages are: loaded and rejected," are stated in the template answer.

The question's answer should clearly evidence knowledge of required salient facts relating to the issues aligned to "depicting what would be the number of page faults if Least Recently Used page-replacement algorithm is used; and depict, in the diagram (in your answer), the order the pages are: loaded and rejected"; this was not done explicitly in some of the answers given.
If the answers did not detail the mechanism plainly as stated in the above (and in the Example answer); e.g. if the answer does not give evidence of knowledge of required salient facts, full marks were not awarded.
The theory that enables you to depict what would be the number of page faults if Least Recently Used page-replacement algorithm is used; and depict, in the diagram (in your answer), the order the pages are: loaded and rejected was covered in the lecture series in three different audio visual media: 1) the actual live lecture; 2) the audio recording of the live lecture; and 3) the real time video of the lecture.

B2.d) A processor has an address bus width of 36 bits. The processor uses this address bus to generate the logical address for a virtual memory system. The computer system in which the processor resides has 1GB of main memory and uses a paged virtual memory system that has a page size of 64KB.

In each case below, please show full working in your answer:

(i) How many pages are there in the paged virtual memory system? (1 mark)

(ii) How many page frames are there in main memory? (1 mark)

(iii) The logical address generated by the processor is made up of two bit fields, a 'page number' (or tag) and an 'offset.' How many bits in the logical address are allocated to the tag (or page number) and how many to the offset? (3 marks)

B2.d)

Application & Bookwork (5 marks),

Example answer:- The following points should be covered to some degree in the answer:

NOTE: with a 1GB of main memory; this implies 30 address lines' or 2^{30} = 1,073,741,824 (or 1GB); while a bus width of 36 bits implies 2^{36} addressable locations [in total] in virtual address space [Logical address space.]. Finally, 64KB page (or 2^{16}) requires 16 address lines.

(i) Number of pages = $\frac{Address\ \ Space}{Page\ \ Size}$ = $2^{36}/2^{16} = 2^{20}$, 1,048,576 (or 1M) pages [in the paged virtual memory system].

1 marks, *1/2 for approach and 1/2 for method.*

(ii) Number of page frames = $\frac{Address\ \ Space}{Block\ \ Size}$ = $2^{30}/2^{16} = 2^{14}$, 16,384 (16K) pages frames [in main memory].

1 marks, *1/2 for approach and 1/2 for method.*

(iii) Need to tag 2^{20} pages, so $\log_2(2^{20})$ = 20 bits for tag.

Offset required to address 64KB, so $\log_2(64K) = \log_2(2^{16})$ = 16 bits for offset.

35, … … ,0

Tag (Page Number)	Offset
20 [1 mark]	16 [1 mark]

3 marks, *2 mark for both result and 1 mark for method [explicitly evidencing correct calculation methodology]. Note the method shown here is not the only one.*

5 marks for a concise description of complete segmentation loading process,
2 ½ marks for half issues covered,
1 mark for some basic understanding (or attempt).

Reference Learning Resources, Background Reading, and Lecture itself for detailed information; Lecture(s) No(s).: 11, Virtual Memory (1).

TOTAL marks (5 marks) [20]

Marker's feedback
Pedagogic assessment [criterion]:
The question assesses Lecture 11, Virtual Memory (1).

Well done, most of you were able to "calculate how many pages are there in the paged virtual memory system; then calculate how many page frames are there in main memory; finally, how many bits in the logical address are allocated to the tag (or page number) and how many to the offset".
The question's answer should clearly evidence knowledge of required salient facts relating to "calculating how many pages are there in the paged virtual memory system; then calculate how many page frames are there in main memory; finally, how many bits in the logical address are allocated to the tag (or page number) and how many to the offset".
In the answer [some of] the following terminology (keywords and naming conventions) should be utilised in context; for example: address; space; logical; page; size; Number of pages; Number of page frames; block; tag; offset; $\log_2$; plus all those underlined in template answer in box above.
Main differentiation that must be clearly evidenced by "calculating how many pages are there in the paged virtual memory system; then calculate how many page frames are there in main memory; finally, how many bits in the logical address are allocated to the tag (or page number) and how many to the offset," are stated in the template answer.
The question's answer should clearly evidence knowledge of required salient facts relating to the issues aligned to "calculating how many pages are there in the paged virtual memory system; then calculate how many page frames are there in main memory; finally, how many bits in the logical address are allocated to the tag (or page number) and how many to the offset"; this was not done explicitly in some of the answers given.
If the answers did not detail the mechanism plainly as stated in the above (and in the Example answer); e.g. if the answer does not give evidence of knowledge of required salient facts, full marks were not awarded.
This theory of calculating how many pages are there in the paged virtual memory system; then calculate how many page frames are there in main memory; finally, how many bits in the logical address are allocated to the tag (or page number) and how many to the offset was covered in the lecture series in three different audio visual media: 1) the actual live lecture; 2) the audio recording of the live lecture; and 3) the real time video of the lecture.

B3.a) An Input/Output (I/O) scheme in a computer system may be controlled by a polled I/O scheme or by an interrupt-driven I/O scheme. In the context of a simple input device (a keyboard) carefully explain how polled and interrupt-driven I/O schemes work. Which would you expect to give the best performance? (6 marks)

B3.a)
Bookwork (1 marks):

Example answer:- The following points should be covered to some degree in the answer:

Polled I/O schemes work:

In this case [polled], the procedure for polling the I/O corresponding to periodically calling a program to interrogate the I/O; which is overseen by a controlling program.
For example, if a character has been typed [entered], it is read and placed in memory. Note a *status* register would be read to check to see if a character had been typed. To place the character in memory a *data* register is read.
If character has not been typed [entered], the program exits and the processor [controlling program] can then get on with useful work.

Interrupt-driven I/O schemes work:

In the case of Interrupt-driven I/O schemes there is no need to read a *status* register, the interrupt indicates that a character has arrived, so we can simply read from the *data* register.
Note that every time a character arrives the processor will be interrupted.

3 marks for an answer that mentions or gives an explanation of polled I/O; must mention keyboard for full marks; and
3 marks for an answer that mentions or gives an explanation of interrupt-driven I/O; must mention keyboard for full marks
3 marks for half issues covered,
1 mark for some basic understanding (or attempt).

Reference Learning Resources, Background Reading, and Lecture itself for detailed information; Lecture(s) No(s).: 14: Controlling Input and Output 1 and Lectures 15: Controlling Input and Output 2.

TOTAL marks (6 marks) [10]

Marker's feedback
Pedagogic assessment [criterion]:
The question assesses Lecture 14: Controlling Input and Output 1 and Lectures 15: Controlling Input and Output 2.

First, I am seriously impressed with the answers given in the exam scripts; and this year's answers are some of the best I have seen, particularly answers to this question.
Well done, most of you were able to "explain how polled and interrupt-driven I/O schemes work and differentiate between then by stating which you would expect to give the best performance?"
The question's answer should clearly evidence knowledge of required salient facts relating to what the "explain how polled and interrupt-driven I/O schemes work and differentiate between then by stating which you would expect to give the best performance?"
In the answer [some of] the following terminology (keywords and naming conventions) should be utilised in context; for example: periodically, calling, program, interrogate, I/O, overseen, controlling program, *status*, register, read, to check, if, character, character, not, typed, program exits, processor, controlling program, get on with useful work, Interrupt-driven I/O, no need, read, *status* register, interrupt, indicates, character, arrived, read, *data* register, every time, character arrives, processor, interrupted.
Main differentiation that must be clearly "evidence and explain how polled and interrupt-driven I/O schemes work and differentiate between then by stating which you would expect to give the best performance?," are stated in the template answer.
The question's answer should clearly evidence knowledge of required salient facts relating to the issues aligned to "explaining how polled and interrupt-driven I/O schemes work and differentiate between then by stating which you would expect to give the best performance?"; this was not done explicitly in some of the answers given.
If the answers did not detail the mechanism plainly as stated in the above (and in the Example answer); e.g. if the answer does not give evidence of knowledge of required salient facts, full marks were not awarded.
The theory that explains how polled and interrupt-driven I/O schemes work and differentiate between then by stating which you would expect to give the best performance? was covered in the lecture series in three different audio visual media: 1) the actual live lecture; 2) the audio recording of the live lecture; and 3) the real time video of the lecture.

B3.b) Direct memory access (DMA) is interrupt driven. Given that a processor writes to a disk, utilizing DMA, describe the four-step DMA process for writing data. (4 marks)

B3.b)
Bookwork (4 marks):

Example answer:- The following points should be covered to some degree in the answer:
A processor writes to a disk; the process can be speeded up utilising direct memory access (DMA).
The 4-steps DMA process is:
1) Processor informs the disk DMA I/O - to write data by writing to a command register in the DMA I/O device;
2) DMA controller starts write process;
3) Process gets on with another work [process] until;
4) DMA device is finished; which is signalled by an interrupt.

4 marks for a concise description of all the four,
2 marks for two,
1 mark for one,
½ marks for some basic understanding (or attempt).
Reference Learning Resources, Background Reading, and Lecture itself for detailed information; Lecture(s) No(s).: 15; Controlling Input and Output 2.
TOTAL marks (4 marks) [4]

Marker's feedback
Pedagogic assessment [criterion]:
The question assesses Lecture 15: Controlling Input and Output 2.

Well done, most of you were able to "describe the four-step DMA process for writing data".
The question's answer should clearly evidence knowledge of required salient facts relating to what the "four-step DMA process for writing data were".
In the answer [some of] the following terminology (keywords and naming conventions) should be utilised in context; for example: Processor, informs, disk, DMA I/O, writing to, command, register, DMA controller, starts, write, process, process, another, process, DMA device, finished, signalled, by an interrupt; plus all those underlined in template answer in box above.
Main differentiation that must be clearly evidenced when "describing the four-step DMA process for writing data" are stated in the template answer.
The question's answer should clearly evidence knowledge of required salient facts relating to the issues aligned to "describe the four-step DMA process for writing data"; this was not done explicitly in some of the answers given.

If the answers did not detail the mechanism plainly as stated in the above (and in the Example answer); e.g. if the answer does not give evidence of knowledge of required salient facts, full marks were not awarded.
This theory of [or behind] the four-step DMA process for writing data was covered in the lecture series in three different audio visual media: 1) the actual live lecture; 2) the audio recording of the live lecture; and 3) the real time video of the lecture.

END OF EXAMINATION