

系統程式

From 線上測驗網站 <http://david.bioinformatic.idv.tw/LawQuiz/>

第 1 章.	作業系統概論.....	10
1.1.	系統架構.....	10
1.1.1.	系統呼叫.....	10
1.1.2.	行程間通訊.....	10
1.1.3.	作業系統的分層方法.....	11
1.1.4.	作業系統的微核心.....	11
1.1.5.	作業系統的模組.....	11
1.1.6.	DMA(Direct Memory Access).....	12
1.2.	作業系統基本觀念.....	12
1.3.	作業系統的演進.....	12
1.4.	多程式系統.....	13
1.5.	分時系統.....	14
1.6.	即時系統.....	14
1.7.	分散式系統.....	14
1.7.1.	分散式系統的種類.....	14
1.7.2.	分散式系統的功能.....	14
1.8.	作業系統資源保護問題.....	14
1.8.1.	輸出入動作的保護.....	14
1.8.2.	記憶體的保护.....	15
1.8.3.	CPU 的保护.....	15
1.9.	中斷.....	16
1.9.1.	中斷與 trap	16
1.9.2.	中斷的種類.....	16
1.9.3.	中斷發生的處理步驟.....	16
1.9.4.	同步中斷與非同步中斷.....	16
1.10.	陽春機器與延伸機器的意義.....	16
1.11.	考古題.....	16
第 2 章.	程序排程.....	17
2.1.	程序基本觀念介紹.....	17
2.2.	排程概念介紹.....	17
2.3.	本文切换(Context Switching).....	17
2.3.1.	Context Switching 的意義.....	17
2.3.2.	PCB(Process Cotrol Block)	17
2.4.	排程方法.....	17
2.4.1.	FCFS	17
2.4.2.	Shortest Job First.....	18
2.4.3.	Preemptive Shortest Job First	18

2.4.4.	Priority schedule	18
2.4.5.	Round-Robin scheduling-RR.....	18
2.4.6.	Multi-level queue	18
2.4.7.	Multi-level feedback queue.....	18
2.5.	Thread	18
2.5.1.	User level thread	18
2.5.2.	kernel level thread.....	18
第 3 章.	記憶體管理.....	20
3.1.	記憶體的層次結構.....	20
3.2.	記憶體保護法.....	20
3.3.	裸機.....	20
3.4.	單一使用者記憶體配置方式.....	20
3.5.	重疊置換(Overlapped Swapping)	20
3.6.	記憶體碎裂(fragmentation)	20
3.7.	固定分割法記憶體管理.....	20
3.8.	變動分割法記憶體管理.....	20
3.8.1.	Merge	20
3.8.2.	Compaction	20
3.9.	多重基底暫存器.....	20
3.9.1.	採用多重基底暫存器的理由.....	20
3.9.2.	作法.....	20
3.9.3.	特性.....	20
3.10.	分頁記憶體管理.....	21
3.10.1.	分頁的意義.....	21
3.10.2.	位址轉換問題.....	21
3.11.	分頁表製作問題.....	22
3.11.1.	直接對應方式.....	22
3.11.2.	關聯對應方式.....	22
3.12.	共用分頁問題.....	22
3.13.	分段記憶體管理.....	22
3.13.1.	使用者的觀點.....	22
3.13.2.	分段的特性.....	23
3.13.3.	分段記憶體管理法邏輯位址的內容.....	23
3.14.	分段表的製作.....	23
3.15.	分頁與分段比較.....	23
3.16.	分頁分段法.....	23
3.17.	分段分頁法.....	23
第 4 章.	虛擬記憶體.....	24

4.1.	虛擬記憶體的基本概念.....	24
4.2.	需求分頁系統(Demand Paging System)	24
4.2.1.	分頁錯誤發生時系統的處理步驟.....	24
4.3.	需求分頁系統的效益評估.....	24
4.4.	分頁取代策略.....	24
4.5.	分頁取代演算法.....	24
4.5.1.	FIFO	24
4.5.2.	最佳取代法(Optimal Algorithm-OPT)	24
4.5.3.	最近最久未再被用到演算法(Least Recently Used-LRU)	24
4.5.4.	二次機會取代法(second chance algorithm)	25
4.5.5.	最少使用次數取代法(Least Frequently Used-LFU).....	25
4.5.6.	最多使用次數取代法(Most Frequently Used-MFU).....	25
4.6.	框的配置問題.....	25
4.6.1.	配置框數的方法.....	25
4.7.	Thrashing	25
4.7.1.	定義.....	25
4.8.	重要觀念與名詞解釋.....	25
4.8.1.	全域取代與區域取代.....	25
4.8.2.	預先分頁.....	26
4.8.3.	I/O interlock	26
4.8.4.	反頁表(Inverted Page Table).....	26
第 5 章.	磁碟管理.....	27
5.1.	磁碟基本觀念介紹.....	27
5.2.	磁碟可用空間管理.....	27
5.2.1.	位元向量.....	27
5.2.2.	可用空間的 linked list.....	27
5.2.3.	Grouping free list	27
5.2.4.	Counting.....	27
5.3.	磁碟的檔案配置.....	28
5.3.1.	continuous	28
5.3.2.	linked.....	28
5.3.3.	indexed	29
5.4.	磁碟排程.....	29
5.4.1.	FCFS	29
5.4.2.	Shortest Seek Time First(SSTF)	29
5.4.3.	SCAN	29
5.4.4.	C-SCAN	29
5.4.5.	Look	29

5.4.6.	C-Look	29
5.5.	磁碟排程法選擇	30
5.6.	大容量儲存裝置	30
5.6.1.	DAS、NAS、SAN	30
5.6.2.	RAID	31
5.7.	啟動區段	31
5.7.1.	bootstrap 儲存在唯讀記憶體(ROM)	32
5.7.2.	只儲存 bootstrap loader 在啟動的 ROM 中	32
第 6 章.	檔案管理	33
6.1.	檔案系統的組織	33
6.1.1.	檔案	33
6.1.2.	目錄	33
6.2.	檔案的操作	33
6.3.	檔案的存取方法	33
6.4.	檔案的保護	34
6.5.	目錄系統	34
6.6.	內部檔案結構	35
第 7 章.	死結	36
7.1.	死結的定義	36
7.2.	死結產生的條件	36
7.3.	死結的預防	36
7.4.	死結的避免	36
7.4.1.	定義	36
7.4.2.	爲了達到「死結避免」的目的，系統所需的三樣資訊	36
7.4.3.	安全狀態	36
7.5.	死結的偵測	37
7.5.1.	死結的偵測	37
7.5.2.	死結偵測的時機	38
7.6.	死結的回復	38
7.6.1.	將產生死結的程序停止其執行動作	38
7.6.2.	資源強占	38
7.7.	飢餓現象	38
第 8 章.	並行程式	39
8.1.	並行敘述	39
8.2.	優先順序圖(Precedence Graph)	39
8.3.	Fork/Join 敘述	39
8.4.	生產者/消費者問題	39
8.5.	競賽條件	39

8.6.	臨界區間.....	39
8.6.1.	臨界區間的意義.....	39
8.6.2.	臨界區間的架構.....	39
8.6.3.	臨界區間的條件(必考).....	39
8.6.4.	Peterson's solution	40
8.6.5.	lock.....	40
8.7.	信號機(Semaphore).....	40
8.7.1.	信號機的作用.....	40
8.7.2.	Signal 和 Wait 的運算.....	41
8.7.3.	解決 Semaphore 忙碌等待的演算法.....	41
8.7.4.	信號機的用途.....	41
第 9 章.	組譯程式.....	43
9.1.	基本概念.....	43
9.1.1.	中央處理單元內部有三個元件.....	43
9.1.2.	運算碼的設定.....	43
9.1.3.	機器週期(Machine Cycle).....	43
9.2.	組合語言的指令種類.....	45
9.2.1.	助憶碼指令(mnemonic instruction).....	45
9.2.2.	假指令(pseudo instruction)	45
9.2.3.	巨集指令(macro instruction).....	45
9.3.	組譯程式的工作.....	45
9.4.	目的碼的內容.....	45
9.4.1.	H-record(標頭紀錄)	45
9.4.2.	T-record(文本紀錄)	45
9.4.3.	E-record(結束紀錄).....	46
9.5.	組譯程式的種類.....	46
9.5.1.	單次處理組譯程式.....	46
9.5.2.	兩次處理組譯程式.....	46
9.5.3.	多次處理組譯程式.....	47
9.6.	資料結構的格式.....	47
9.6.1.	靜態表格.....	47
9.6.2.	動態表格.....	47
9.7.	程式重定位.....	48
9.7.1.	程式重定位的定義.....	48
9.7.2.	絕對程式.....	48
9.7.3.	可重定址程式.....	48
9.7.4.	修飾紀錄(Modification record).....	48
9.8.	程式區段.....	48

9.9.	控制段.....	48
9.9.1.	控制段的意義.....	48
9.9.2.	外部參考.....	48
9.9.3.	指令說明.....	48
9.9.4.	重要觀念.....	49
9.10.	具有 Overlay 結構的兩次處理組譯程式.....	49
9.11.	常見的定址模式.....	49
9.11.1.	立即定址模式.....	49
9.11.2.	直接定址模式.....	49
9.11.3.	間接定址模式.....	50
9.11.4.	索引定址模式.....	50
9.11.5.	基底定址模式.....	50
9.11.6.	相對定址模式.....	50
9.12.	考古題.....	50
第 10 章.	鏈結載入程式(linking loader).....	52
10.1.	載入程式的定義與功能.....	52
10.1.1.	載入程式的定義.....	52
10.1.2.	載入程式的功能.....	52
10.2.	重新定址(relocation).....	52
10.2.1.	基本觀念.....	52
10.2.2.	重定址載入程式.....	52
10.2.3.	重定址製作法.....	52
10.3.	連結載入程式(Linking Loader)的製作.....	53
10.3.1.	為什麼要做連結.....	53
10.3.2.	實例說明.....	53
10.3.3.	製作法.....	54
10.3.4.	多個修飾紀錄的問題.....	54
10.4.	一般載入程式.....	55
10.4.1.	一般載入程式的作法.....	55
10.4.2.	一般載入程式的優點.....	55
10.5.	Assembler and Go Loader	55
10.5.1.	Assembler and Go Loader 的作法	55
10.5.2.	優點.....	55
10.5.3.	缺點.....	55
10.6.	絕對載入程式.....	55
10.6.1.	絕對載入程式的功能.....	55
10.6.2.	優點.....	55
10.6.3.	缺點.....	55

10.7.	直接連結載入程式.....	55
10.7.1.	直接連結載入程式的功能.....	55
10.7.2.	目的碼的資訊.....	55
10.7.3.	DLL 的兩次處理.....	56
10.7.4.	DLL 的優點.....	56
10.7.5.	DLL 的缺點.....	56
10.8.	BSS Loader (Binary Symbolic Subroutine Loader)	56
10.8.1.	BSS 的功能	57
10.8.2.	BSS loader 特性介紹	57
10.8.3.	BSS 缺點	57
10.9.	連結編輯程式.....	57
10.9.1.	連結編輯程式的作法.....	57
10.9.2.	連結編輯程式的優點.....	57
10.10.	自動程式庫搜尋.....	57
10.11.	載入程式的功能選擇.....	57
10.12.	覆疊程式(Overlay Program)	57
10.13.	動態連結.....	57
10.13.1.	基本觀念.....	57
10.13.2.	動態連結的意義.....	57
10.13.3.	動態連結的製作.....	57
10.13.4.	DLL 的優缺點.....	58
10.14.	考古題.....	58
第 11 章.	巨集處理程式.....	59
11.1.	基本觀念.....	59
11.2.	巨集的基本概念介紹.....	59
11.2.1.	巨集.....	59
11.2.2.	巨集定義.....	59
11.2.3.	巨集展開.....	59
11.3.	巨集與副程式的區別.....	59
11.4.	巨集中參數處理的方式.....	59
11.4.1.	巨集指令引數.....	59
11.4.2.	巨集參數處理的方式有以下兩種.....	60
11.5.	巨集定義與巨集呼叫間的關係.....	60
11.5.1.	在巨集中呼叫巨集.....	60
11.5.2.	巨集定義內含巨集定義.....	61
11.6.	巨集中標記(Label)的處理方式.....	62
11.6.1.	Label 重複出現的基本觀念	62
11.6.2.	解決辦法.....	62

11.7.	條件式巨集展開.....	62
11.8.	巨集處理程式的製作.....	62
11.8.1.	基本的資料結構.....	62
11.8.2.	範例.....	62
11.9.	巨集組譯程式.....	62
第 12 章.	語法與編譯程式.....	64
12.1.	編譯程式基本概念.....	64
12.2.	P-碼編譯器(P-code compiler).....	64
12.3.	Compiler's compiler	64
12.4.	交互式編譯程式.....	64
12.5.	自動機(Automata)理論	64
12.5.1.	自動機.....	64
12.5.2.	有限自動機(Finite Automata-FA).....	64
12.5.3.	DFA(Deterministic Finite Automata)	64
12.5.4.	NFA(Nondeterministic Finite Automata)	65
12.6.	語法的基本定義.....	65
12.7.	文法的要素(組成要素).....	65
12.8.	文法的分類(Classification of grammers).....	65
12.8.1.	Type 0：無任何限制。	65
12.8.2.	Type 1：與上下文相關的文法.....	65
12.8.3.	Type 2：與上下文無關的文法 (context-free grammar)	65
12.8.4.	Type 3：正規文法 (regular grammar)	65
12.9.	Backus Naur Form(BNF 文法).....	66
12.10.	剖析樹(parse tree)	66
12.11.	模擬兩可的文法(ambiguous grammar).....	66
12.12.	描述程式語言語法的方式.....	66
12.13.	剖析法(Parsing).....	66
12.13.1.	Top down	66
12.13.2.	Bottom up.....	66
12.14.	最佳化(Optimization).....	67
12.14.1.	與機器相關的最佳化.....	67
12.14.2.	與機器無關的最佳化.....	67