

國立臺灣科技大學答案卷

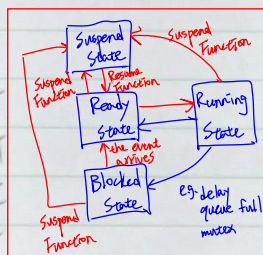
National Taiwan University of Science and Technology Answer Sheet

姓名/Name 張恒豪 學號/Student ID B11002110 班級/Class 四電子三乙
 科目/Course title 嵌入式系統設計 日期/Date 112.11.29

評分 Score	教師簽章 Signature of Lecturer
97	

記分欄 從此處開始寫起。試卷用紙務須節用，非經主試認可不得續用其他紙張作答。/Please write from here.

沒拿到B卷是非題題目，所以在此附上A卷是非題題目及參考詳解：



1. (F) FreeRTOS is a class of RTOS that is designed to be small enough to run on a microcontroller, it allocates memory with malloc() and free the memory with free().
2. (T) Gatekeeper tasks provide a clean method of implementing mutual exclusion without the risk of priority inversion or deadlock.
3. (T) FreeRTOS uses TickType_t to measure the time, it can be either an unsigned 16-bit type, or an unsigned 32-bit type, depending on the setting in FreeRTOSConfig.h.
4. (F) In FreeRTOS, the kernel objects such as tasks, queues, semaphores and event groups are not dynamically allocated at run-time, but statically allocated at compile-time.
5. (F) Heap_2 uses a first fit algorithm to allocate memory.
6. (F) Heap_3 uses pvPortMalloc() and vPortFree() to allocate and free the memory.
7. (F) In FreeRTOS task management, it is possible to have two or more tasks in the running state simultaneously.
8. (T) The Blocked state can be changed to Ready state by an event, but the Ready state will not be changed to Blocked state in any way.
9. (T) Any state can be changed to the Suspended state, but the Suspended state can only be changed to the Ready state.
10. (T) Heap_4 uses a first fit algorithm to allocate memory.
11. (F) Heap_5 uses a best fit algorithm to allocate memory.
12. (T) The parameters to vTaskDelayUntil() specify the exact tick count value at which the calling task should be moved from the Blocked state into the Ready state.
13. (T) It is possible to add application specific functionality into the idle task through the use of an idle task hook function, which is called automatically by the idle task.
14. (T) FreeRTOS defaults to using Fixed Priority Preemptive Scheduling with Time Slicing to schedule tasks.
15. (F) In Fixed Priority Preemptive Scheduling, once a task's priority is set, it cannot be modified anymore.
16. (F) In Co-operative Scheduling, lower priority tasks may be preempted by higher priority tasks.
17. (T) A queue is a data storage that can hold a finite number of fixed size data items, which provides a task-to-task communication mechanism in FreeRTOS.
18. (T) Non-reentrant function cannot be accessed from more than one task, or it may suffer from the risk of data or logical operation corruption.
19. (F) Recursive mutex occurs when two tasks cannot proceed because they are both waiting for a resource that is held by the other.
20. (F) When using a gatekeeper task to manage shared resources, other high-priority tasks can directly access the shared resources if they are available.

21. The tasks in FreeRTOS are managed by classifying them into the four states: Running state, Blocked state, Suspended state, and Ready state.
- (a) A task that is waiting for an event is said to be in the "Blocked State". Tasks can enter the Blocked state to wait for the Temporal events or Synchronization events. Explain what is Temporal events/Synchronization events.
- (b) What does that mean if a task is in the Ready state?

21.(a) Temporal event 發生固定數量的 tick interrupts 到來時，呼叫 vTaskDelay 的 task 會被 scheduler 移到 Blocked state 直到 Temporal event 發生。

Synchronization event 發生在資料的一致性可被保證或欲存取的资源已被使用時。當一個沒有取得 mutex 的 task 在要存取需要 mutex 才能存取的資料時，此 task 就會被移動到 block state 來等待 Synchronization event 發生。當一個 task 要寫入 queue，但 queue 已經滿了時，此 task 會被移到 Blocked state 來等待 Synchronization event 發生 (queue 有空間) 或是 Temporal event 發生 (超過設定的當 queue 滿時等待的最大 ticks 數量)。

21.(b) 沒有在 running，也沒有被 suspend 或 block。

22. FreeRTOS has five kinds of memory allocation schemes, Heap_1, Heap_2, Heap_3, Heap_4, and Heap_5.

- (a) Why Heap_1 do not have to consider the more complex memory allocation issues, such as determinism and fragmentation?
- (b) What algorithm does Heap_2 use to allocate memory? Explain the main idea of this algorithm.
- (c) What algorithm does Heap_4 use to allocate memory? Explain the main idea of this algorithm.

記分欄

轉頁從此開始寫起。

22. (a) 因為 Heap_1 不會在執行過程中 free 或 create heap。所有記憶體配置都在應用程式執行前就完成了，且這些配置好的記憶體空間永遠不會被釋放，直到斷電或重新啟動。
22. (b) Heap_2 使用 best fit algorithm 來 allocate memory。此算法會選擇比現在要 allocate 的空間要大且大小也最相近的空間記憶體區塊來 allocate。當釋放記憶體空間後，相鄰的空間記憶體區塊並不會被合併。
22. (c) Heap_4 使用 first fit algorithm 來 allocate memory。此算法會循序搜尋空間的記憶體空間，並選則第一個能容納現在要 allocate 的空間大小的空間記憶體區塊來 allocate。當釋放記憶體空間後，相鄰的空間記憶體區塊會被合併。

23. Explain what is Priority Inversion? How FreeRTOS mitigates the impact of Priority Inversion?

23. Priority Inversion 是高優先級的 task 因為沒有拿到 mutex 而需要等待低優先級的 mutex holder task 執行完才能被執行。如果這個低優先級的 mutex holder task 還被其他中等優先級的 task 搶佔的話，那就得等到中等優先級的 task 進 block 或被處理完，再等低優先級的 mutex holder 也被執行完後這個高優先級的 task 才能拿到 mutex 並開始執行。

FreeRTOS 透過 priority inheritance 來減輕 Priority Inversion 所造成的影響。持有 mutex 的 task 的優先級會被提升到與需要 mutex 的 task 同樣高的優先級來避免 mutex holder 被中等優先級的 task 搶佔，解決 unbounded priority 的隱憂。

此舉

23. FreeRTOS uses Prioritized Preemptive Scheduling with Time Slicing to schedule tasks.

- (a) Explain what is "Time Slicing"?
- (b) What will happen if we do not use time slicing to schedule the tasks in FreeRTOS?

24. (a) 多個同等優先級的 task 會透過輪流進入 running state 來平分被處理器執行的時間。
24. (b) 當有多個同等優先級的 task 要被 schedule 時，會只有一個 task 能持續多個 tick period 處在 running state，直到它被 block 或自願退出 running state，其他同等優先級的 task 才能進到 running state。

25. (Bonus) What is the name of the professor teaching this class?

25. 陳奕伸。

+2