Esame 1 appello architettura s.o

1)      1) Nell'ambito dei mutex se l'istruzione di Test & Set Lock dell'ARM è TST ed il codice assembly per realizzare il lock del mutex il seguente:

mutex\_lock: TST R0, MUTEX

CMP R0, #0

BEQ fine

<istruzione mancante>

B mutex\_lock

fine: BX chiamante

Quali istruzioni si possono utilizzare?

- MOV MUTEX, #0

- BL thread\_yield

- non occorre alcuna istruzione è corretto già così

- CMP R0, #0

- MOV R0, #0   VERA

Risposta:

La risposta corretta sarebbe "MOV R0, #0" in quanto questa istruzione imposterà il valore della locazione di memoria "MUTEX" come 0, indicando che il mutex è stato acquisito. "MOV MUTEX, #0" non è corretto perché non è una istruzione di test-and-set. "BL thread\_yield" non è corretto perché non è un'istruzione per acquisire il lock del mutex. "non occorre alcuna istruzione è corretto già così" non è corretto perché occorre una istruzione per acquisire il lock del mutex. "CMP R0, #0" non è corretto perché non è un'istruzione per acquisire il lock del mutex.

2) Descrivere l'evoluzione delle architetture multilivello partendo dall'alto (più antica) verso il basso (più recente)

1 Invenzione della macchina analitica

2 Invenzione della microprogrammazione

3 Invenzione del sistema operativo

4 Migrazione delle funzionalità verso il microcodice

5 Eliminazione della microprogrammazione

risposta:

1 Invenzione della macchina analitica

2 Invenzione del sistema operativo

3 Invenzione della microprogrammazione

4 Migrazione delle funzionalità verso il microcodice

5 Eliminazione della microprogrammazione

3) ALGEBRA DI BOOLE - Quali circuiti permettono di scrivere qualsiasi funzione logica\

- Demultiplexer

- PLA

- Multiplexer

- Porte universali

- Decoder

Risposta:- Demultiplexer- PLA- Multiplexer- Porte Universali- Decoder.

4) SISTEMI OPERATIVI - Descrivere l’algoritmo ottimo di sostituzione delle pagine indicando la sequenza partendo dall'alto verso il basso

1 accade un page-fault

2 si etichetta ogni pagina caricata con il numero di istruzioni che sarà eseguito prima che quella pagina sia referenziata.

3 per ogni pagina caricata si calcola il numero di istruzioni che sarà eseguito prima che quella pagina sia referenziata

4 si scorrono tutte le pagine ricercando quella con etichetta massima

5 si sostituisce la pagina che ha l'etichetta più alta.

risposta:

1 accade un page-fault

2 si etichetta ogni pagina caricata con il numero di istruzioni che sarà eseguito prima che quella pagina sia referenziata utilizzando un algoritmo di previsione come l'algoritmo di previsione di Belady.

3 per ogni pagina caricata si calcola il numero di istruzioni che sarà eseguito prima che quella pagina sia referenziata utilizzando l'algoritmo di previsione scelto

4 si scorrono tutte le pagine presenti in memoria ricercando quella con etichetta massima

5 si sostituisce la pagina che ha l'etichetta più alta con la pagina richiesta dalla page-fault

Questo algoritmo è noto come Algoritmo di Sostituzione delle Pagine Optimal (OPT) o Algoritmo di Sostituzione delle Pagine di Belady poiché utilizza un algoritmo di previsione per etichettare ogni pagina e sceglie sempre la pagina con l'etichetta più alta per sostituirla.

5)HAMMING : Supponendo di voler trasmettere la sequenza binaria corrispondente al numero esadecimale 1E01 (esercizio precedente), se arriva la sequenza esadecimale 036805 quale sarà il valore dei bit di controllo calcolati?

risposta: 01100

6) ARCHITETTURE DEI CALCOLATORI - Nell'ambito della codifica dei caratteri quali affermazioni non sono vere? Indicare TUTTE le risposte corrette, questa domanda può avere da 0 a 5 risposte corrette

- UNICODE non è in grado di gestire l'ordinamento degli ideogrammi nella mappa

- UNICODE può codificare la nascita di nuovi ideogrammi

- Nel codice ASCII i codici iniziali sono caratteri di controllo

- UTF-8 utilizza una codifica fissa

- UTF-8 utilizza un codice auto-sincronizzante

risposta:

Le risposte corrette sono:

* UNICODE non è in grado di gestire l'ordinamento degli ideogrammi nella mappa (Falso, UNICODE include anche un ordinamento canonico per gli ideogrammi)
* UNICODE può codificare la nascita di nuovi ideogrammi (Vero, UNICODE è in grado di codificare nuovi ideogrammi e simboli man mano che vengono creati)
* Nel codice ASCII i codici iniziali sono caratteri di controllo (Vero, i codici ASCII tra 0 e 31 sono caratteri di controllo)
* UTF-8 utilizza una codifica fissa (Falso, UTF-8 utilizza una codifica variabile a lunghezza, ovvero utilizzando un numero variabile di byte per rappresentare un singolo carattere)
* UTF-8 utilizza un codice auto-sincronizzante (Vero, UTF-8 utilizza un codice auto-sincronizzante che consente di determinare la lunghezza di un carattere anche se non si dispone di un carattere di fine riga).

7)SISTEMI OPERATIVI - quale condizioni non sono necessarie per evitare le corse critiche?

-Nessun processo può eccedere il suo quantum di tempo V

-Nessun processo in esecuzione al di fuori della sua regione critica può bloccare altri processi F

-Se un processo è più veloce degli altri può mantenere assegnate le risorse V

-Due processi non possono rimanere all'interno delle loro regioni critiche allo stesso tempo F

-Nessun processo deve restare in attesa infinita per poter entrare nella sua regione critica V

RISPOSTA:

* Nessun processo può eccedere il suo quantum di tempo (Questa condizione riguarda la gestione dei processi e non è strettamente necessaria per evitare le corse critiche)
* Se un processo è più veloce degli altri può mantenere assegnate le risorse (Questa condizione riguarda l'equità nell'assegnazione delle risorse e non è strettamente necessaria per evitare le corse critiche)
* Nessun processo deve restare in attesa infinita per poter entrare nella sua regione critica (Questa condizione riguarda la gestione dei processi e non è strettamente necessaria per evitare le corse critiche, un meccanismo di gestione della coda di attesa può essere utilizzato per gestire i processi in attesa)

8) SISTEMI OPERATIVI - Nell'ambito della gestione dei processi quali affermazioni non sono vere?

-Tutte le operazioni in un semaforo sono atomiche V

-La soluzione di Peterson non richiede busy waiting F

-La soluzione di Peterson non risente del problema dell’inversione delle priorità F

-L'istruzione TSL fa sprecare del tempo di CPU perché richiede busy waiting V

-Nel caso ci siano più CPU occorre proteggere il semaforo con istruzioni TSL (o XCHG) V

Le risposte corrette sono:

* Tutte le operazioni in un semaforo non sono atomiche (Vero, la modifica di un semaforo potrebbe richiedere più di un'istruzione, non è atomica)
* La soluzione di Peterson non richiede busy waiting (Falso, la soluzione di Peterson richiede busy waiting per evitare la soluzione della corsa critica)
* La soluzione di Peterson non risente del problema dell’inversione delle priorità (Falso, la soluzione di Peterson può risentire del problema dell’inversione delle priorità, se un processo con priorità più alta è bloccato da un processo con priorità più bassa)
* L'istruzione TSL fa sprecare del tempo di CPU perché richiede busy waiting (Vero, l'istruzione TSL richiede busy waiting e quindi può comportare uno spreco di tempo di CPU)
* Nel caso ci siano più CPU occorre proteggere il semaforo con istruzioni TSL (o XCHG) (Vero, in caso di più CPU occorre utilizzare istruzioni TSL o XCHG

9) CODICE DI HAMMING - Supponendo di voler trasmettere la sequenza binaria corrispondente al numero esadecimale 1E01, quali sono i bit di controllo da aggiungere alla parola se si utilizza il codice di Hamming? La risposta è nella forma b4 b3 b2 b1 b0 in coerenza con il libro di esercizi e di testo.

Risposta: 00001

10) SISTEMI OPERATIVI - Nell'ambito degli algoritmi di sostituzione delle pagine quali affermazioni non sono vere?

- FIFO non può lavorare con politiche di sostituzione delle pagine locali

- WSClock può anche lavorare con politiche di sostituzione delle pagine locali

- LRU può lavorare anche con politiche di sostituzione delle pagine globali.

- Working Set può lavorare esclusivamente con politiche di sostituzione delle pagine locali

- Tutte le approssimazioni dell’LRU possono lavorare solo con politiche di sostituzione delle pagine globali.

Risposta:

Le affermazioni non vere sono:

* FIFO non può lavorare con politiche di sostituzione delle pagine locali (Falso, FIFO può lavorare con politiche di sostituzione delle pagine locali, come ad esempio l'algoritmo "FIFO per gruppi di pagine")
* LRU può lavorare anche con politiche di sostituzione delle pagine globali (Vero, LRU può lavorare con politiche di sostituzione delle pagine globali)
* Tutte le approssimazioni dell’LRU possono lavorare solo con politiche di sostituzione delle pagine globali (Falso, alcune approssimazioni dell'LRU, come ad esempio l'algoritmo "LRU per gruppi di pagine", possono lavorare con politiche di sostituzione delle pagine locali)

11) ASSEMBLY ARM - Supponendo che il registro R0 contenga il valore 0x000000E8 con quale istruzioni si può inserire in R1 il valore 0x00E800E8?

- LDR R1,=0x00E800E8

- MOV R1, 0x00E800E8

- ADD R1, R0, R0, LSL #16

- PKHBT R1, R0, R0, LSL #16

- UXTB16 R1, R0

Le istruzioni valide per inserire il valore 0x00E800E8 in R1 partendo dal valore contenuto in R0 (0x000000E8) sono:

* ADD R1, R0, R0, LSL #16 (questa istruzione effettua una operazione di shift left logico di 16 bit su R0 e lo somma a se stesso, inserendo il risultato in R1)
* PKHBT R1, R0, R0, LSL #16 (questa istruzione effettua una operazione di shift left logico di 16 bit su R0 e lo combina con R0 stesso, inserendo il risultato in R1)

Le altre istruzioni non sono valide in questo caso:

* LDR R1,=0x00E800E8 (questa istruzione carica un valore costante nell'indirizzo specificato in R1)
* MOV R1, 0x00E800E8 (questa istruzione copia un valore costante in R1)
* UXTB16 R1, R0 ( questa istruzione estende un byte a 16 bit, non è adatta per questo caso)

12) ARCHITETTURE DEI CALCOLATORI - Nell'ambito dei dispositivi RAID quali affermazioni non sono vere?

- In un RAID1 le prestazioni in scrittura sono le stesse di un sistema SLED

- Il RAID livello 2 utilizza i byte per decomporre le informazioni sui vari dischi

- RAID0 ha maggiore affidabilità di un sistema SLED con medesimo MTBF

- Il RAID livello 3 è una versione semplificata del RAID 2

- Nel RAID4 le prestazioni sono alte se si aggiornano piccole quantità di dati

Le affermazioni non vere sono:

* In un RAID 1 le prestazioni in scrittura sono le stesse di un sistema SLED (Falso, in un RAID1 le prestazioni in scrittura sono inferiori a quelle di un sistema SLED poiché i dati devono essere scritti su entrambi i dischi)
* Il RAID livello 2 utilizza i byte per decomporre le informazioni sui vari dischi (Falso, Il RAID livello 2 utilizza i bit per decomporre le informazioni sui vari dischi)
* RAID0 ha maggiore affidabilità di un sistema SLED con medesimo MTBF (Falso, RAID0 non fornisce alcuna protezione contro la perdita di dati, quindi la sua affidabilità è inferiore a quella di un sistema SLED)
* Il RAID livello 3 è una versione semplificata del RAID 2 (Falso, Il RAID livello 3 è un RAID che utilizza la parità distribuita su più dischi, non ha nulla a che fare con il RAID 2)
* Nel RAID 4 le prestazioni sono alte se si aggiornano piccole quantità di dati (Vero, Nel RAID4 le prestazioni sono alte se si aggiornano piccole quantità di dati poiché solo un disco

13) ARCHITETTURE DEI CALCOLATORI - Nell'ambito dei circuiti elettronici quali affermazioni sono false?

* Posso collegare insieme le uscite di più porte logiche se sono open collector
* lo stato di alta impedenza delle porte tri-state consente il collegamento delle uscite insieme
* Posso collegare l'uscita di una porta logica agli ingressi di altre
* Posso collegare insieme le uscite di più porte logiche
* teoricamente potrei collegare le uscite insieme se assumessero lo stesso valore (0 o 1)

Risposte:

Le affermazioni false sono:

1. Posso collegare insieme le uscite di più porte logiche se sono open collector (non è sempre possibile, dipende dalla configurazione dell'open collector)
2. Posso collegare l'uscita di una porta logica agli ingressi di altre (non è consigliabile in quanto può causare un cortocircuito)
3. Posso collegare insieme le uscite di più porte logiche (non è possibile in quanto può causare un cortocircuito)
4. teoricamente potrei collegare le uscite insieme se assumessero lo stesso valore (0 o 1) (non è possibile in quanto può causare un cortocircuito)

14) SISTEMI OPERATIVI - Nella gestione dei dispositivi di I/O quali non sono svantaggi del port-mapped I/O

È necessario un meccanismo di protezione speciale per controllare lo svolgimento delle operazioni di I/O da parte dei processi utente F

Il sistema operativo non riesce ad assegnare in modo semplice e dinamico i dispositivi ai processi utente F

I registri di controllo del device sono visti come variabili in memoria F

Per leggere e scrivere i registri di controllo del dispositivo, sono necessarie istruzioni dedicate assembly: IN e OUT F

I driver di controllo dei dispositivi possono essere scritti utilizzando solo il linguaggio C V

15) ALGEBRA DI BOOLE - In quale dei seguenti modi si può scrivere la funzione descritta dalla mappa k ?

not(A)not(C)+not(B)not(D)+BC+Cnot(D) V

not(A)not(C)+BC+Cnot(D) F

not(B)not(D)+not(A)+Cnot(D)+CB+not(A)not(C) F

not(A)B+Cnot(D)+CB+not(A)not(C) F

not(B)not(D)+not(A)B+Cnot(D)+CB+not(A)not(C) V