

1. Descrivere l'architettura interna del M68000.
2. Illustrare il formato istruzione del M68000 e indicare quanti accessi in memoria sono necessari per la fase di fetch e per la fase di execute per la seguente istruzione:
move.B #1, D0
3. Illustrare il concetto di automa a stati finiti e scrivere il grafo di un automa che presenta in uscita 0 o 1 a seconda che gli ultimi 4 bit ricevuti in ingresso contengano un numero di 1 pari o dispari.
4. Descrivere il concetto di memoria cache.
5. Illustrare il formato istruzione del M68000 e indicare quanti accessi in memoria sono necessari per la fase di fetch e per la fase di execute per la seguente istruzione:
move.W #257, D0
6. Illustrare il modello di Von Neumann e la differenza con il modello Harvard.
7. Illustrare cosa è uno stack e il suo utilizzo per lo scambio di parametri con una subroutine .
8. Scrivere il codice assembler che calcola la media di **n** numeri codificati su word
9. Illustrare il formato istruzione del M68000 e indicare quanti accessi in memoria sono necessari per la fase di fetch e per la fase di execute per la seguente istruzione:
move.L #1, D0
10. Illustrare il ciclo di Von Neumann e il ruolo delle interruzioni.
11. Illustrare differenza tra subroutine e istruzione di salto condizionato.
12. Scrivere il codice assembler che calcola trasforma maiuscole e in minuscole in un array di caratteri
13. Illustrare il formato istruzione del M68000 e indicare quanti accessi in memoria sono necessari per la fase di fetch e per la fase di execute per la seguente istruzione:
move.L D0, D0
14. Descrivere il modello astratto di periferica e un esempio di gestione dell'hardware in easy68k
15. Illustrare il concetto di interruzione e la sua gestione nel M68000
16. Illustrare il concetto di memoria virtuale.
17. Illustrare il formato istruzione del M68000 e indicare quanti accessi in memoria sono necessari per la fase di fetch e per la fase di execute per la seguente istruzione:
move.B \$8000, D0
18. Illustrare le differenze tra l'architettura Intel e l'architettura Motorola68000
19. Descrivere la differenza tra direttiva di assemblaggio e istruzione. Elencare le direttive note.
20. Illustrare la differenza tra una macchina combinatoria e una macchina sequenziale, con esempi.
21. Illustrare il formato istruzione del M68000 e indicare quanti accessi in memoria sono necessari per la fase di fetch e per la fase di execute per la seguente istruzione:
move.L \$8000, D0
22. Illustrare la differenza tra architetture CISC e RISC.
23. Descrivere la differenza tra un sistema embedded e un sistema general purpose, con esempi.
24. Descrivere i modi di indirizzamento del motorola 68000, con esempi.
25. Illustrare il formato istruzione del M68000 e indicare quanti accessi in memoria sono necessari per la fase di fetch e per la fase di execute per la seguente istruzione:
move.L (a0), D0
26. Illustrare il compito e le tecniche di realizzazione dell'unità di controllo di un processore.
27. Illustrare la differenza tra memoria RAM, memoria cache e memoria di massa.
28. Scrivere le istruzioni assembler per verificare la presenza di un byte in un array.
29. Illustrare il formato istruzione del M68000 e indicare quanti accessi in memoria sono

necessari per la fase di fetch e per la fase di execute per la seguente istruzione:

move.W (a0), VAR

30. Spiegare la differenza tra un codificatore e un multiplexer.
31. Descrivere le direttive assembler del M68000 per la dichiarazione di variabili in memoria.
32. Illustrare il concetto di memoria virtuale.
33. Illustrare il formato istruzione del M68000 e indicare quanti accessi in memoria sono necessari per la fase di fetch e per la fase di execute per la seguente istruzione:
move.W (a0), (a1)
34. Illustrare il ruolo del registro di stato e la sua organizzazione nel M6800.
35. Illustrare I metodi per il passaggio di parametri nelle subroutines.
36. Illustrare il metodo di gestione autovettorizzato delle interruzioni.