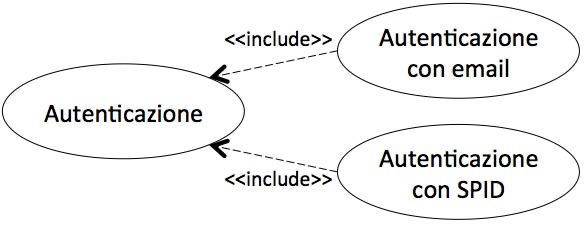
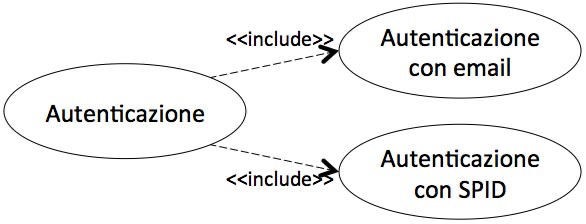
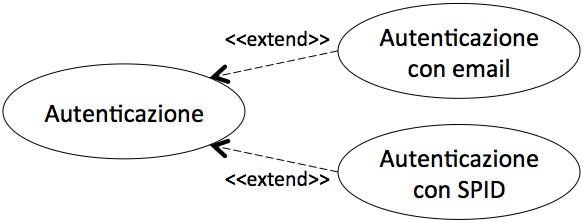
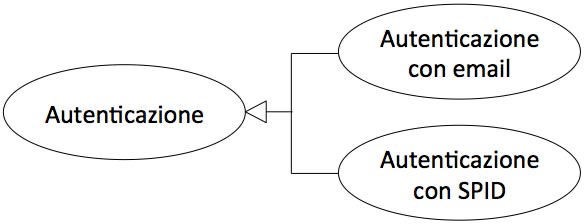
# Compito di IS del 11/1/2021

1. Com’è classificata la *riusabilità* di un software?
2. Esterna e di Processo
3. Interna e di Prodotto
4. Esterna e di Prodotto
5. Interna e di Processo
6. Quali di questi diagrammi UML vengono normalmente utilizzati a supporto del *collaudo* del software, con particolare riferimento al “testing in the large”?
7. Sequenza
8. Attività
9. Classi
10. Casi d'uso
11. Quali di queste *tecniche di verifica* sono applicabili in un contesto black-box?
12. Code walk-through
13. Testing in the small
14. Code inspection
15. Testing in the large
16. nessuna delle precedenti
17. Un software viene modificato per aumentare l'efficienza di una funzionalità di importazione massiva di dati. Di che tipo di *manutenzione* si tratta?
18. Correttiva
19. Evolutiva
20. Perfettiva
21. Adattiva
22. Quali di questi *modelli* sono specificamente pensati per situazioni progettuali in cui i requisiti utente sono incerti o mutevoli nel tempo?
23. Modello MDA
24. Modello RAD
25. Modello a cascata
26. Nessuno dei precedenti
27. Quali delle seguenti affermazioni, riguardanti le funzioni di tipo dato e tipo transazione, sono vere?
28. La logica di un processo EQ (External Inquiry) non contiene formule matematiche o calcoli e non crea dati derivati
29. Il compito primario di un EIF (External Interface File) è di contenere dati referenziati da uno o più processi elementari dell’applicazione che si sta contando
30. Il compito principale di un EO (External Output) è di modificare il comportamento del sistema
31. Nessuna delle precedenti
32. Le special GUI sono pensate per situazioni in cui...
33. ...l'utente deve effettuare un data-entry massivo
34. ...l'utente può avere scarsa esperienza con l'utilizzo dei PC
35. ...la soddisfazione dell'utente nell'uso dell'interfaccia è importante
36. ...memorabilità e apprendibilità dell'interfaccia non sono importanti
37. nessuna delle precedenti
38. L'autenticazione a un portale si può effettuare tramite email oppure tramite lo SPID (Sistema Pubblico d'Identità Digitale). Scegliere tra le seguenti soluzioni di modellazione quella più corretta.



a.

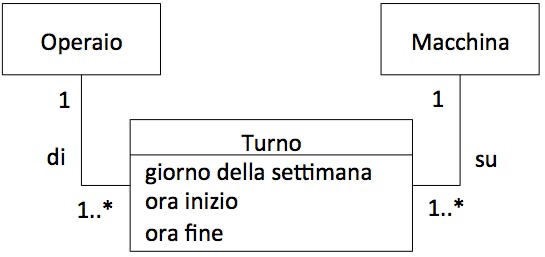
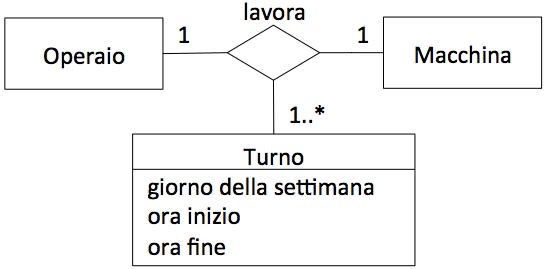
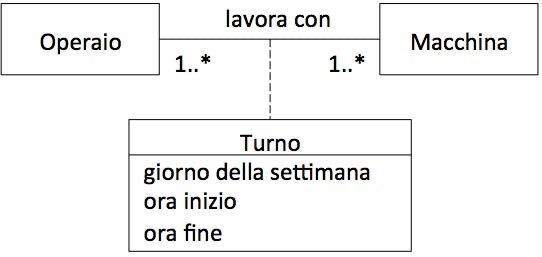
b.

c.

d.

# Compito di IS del 15/2/2021

1. Un operaio effettua turni settimanali di lavoro alle macchine. Durante ogni turno ciascun operaio lavora a una singola macchina, e ogni macchina è manovrata da un singolo operaio. Scegliere, tra le seguenti soluzioni di modellazione, quella ritenuta più appropriata.



a.

b.

c.

1. Cosa si intende per “messa in produzione” del software?
2. il momento in cui inizia la fase di analisi dei requisiti
3. il momento in cui inizia la fase di esercizio del software
4. il momento in cui inizia la fase di implementazione
5. il momento in cui inizia la fase di diagnosi e manutenzione
6. il momento in cui inizia la fase di progettazione del software
7. il momento in cui inizia la fase di collaudo
8. il momento in cui il software viene installato
9. Quali tra i seguenti sono meccanismi di *estendibilità* in UML?
10. specifiche
11. viste
12. profili
13. diagrammi
14. proprietà
15. stereotipi
16. E' dato il seguente frammento di pseudocodice:

begin

read(A,B,C)

if (A>0) AND (B<0) then C := C/(A+B)

end

Quali dei seguenti insiemi di test soddisfano il *criterio di copertura delle decisioni*?

a. {(A=2, B= –1, C=3), (A= –1, B=2, C=0)}

b. {(A=2, B=1, C=0), (A=-1, B= –1, C=3)}

c. {(A=2, B= –1, C=3)}

d. {(A=2, B= –1, C=3), (A= –1, B= –1, C=3)}

1. Cosa si intende per *affordance* in ambito interfacce utente?
2. il posizionamento strutturato di testo, disegni e controlli all’interno di un’area grafica considerata
3. gli aspetti grafici di un oggetto che invitano l'utente a interagire con quell'oggetto in un certo modo
4. una figura che illustra un oggetto attraverso una somiglianza o un’analogia con un concetto del mondo reale
5. Quali tra le seguenti tecniche possono essere ricondotte al *principio di separazione degli argomenti*?
6. la capacità di prevedere i cambiamenti a cui il software sarà sottoposto durante il suo ciclo di vita
7. la ricerca di un problema noto che generalizzi il problema da risolvere
8. l'utilizzo di formalismi e metodologie standard
9. l'identificazione, nel ciclo di vita, di attività distinte e temporalmente correlate
10. l'identificazione di moduli distinti e interconnessi all'interno del software

# Compito di IS del 9/6/2021

1. Dato il seguente frammento di pseudocodice, se ne calcoli la *complessità ciclomatica*: Risposta: 4?

begin

read(N);

read(x);

for i:=1 to N do read(V[i]);

i:=1;

K:=0;

while (i<N) AND (K<100) do begin

if (x<>0) V[i]:=(V[i]+V[i+1])/x;

K:=K+V[i]; i:=i+1;

end end

1. Dati i due seguenti frammenti di diagrammi degli stati, dire se modellano lo stesso comportamento al variare di temperatura e pressione.

pressione>=100 [temperatura>=200]

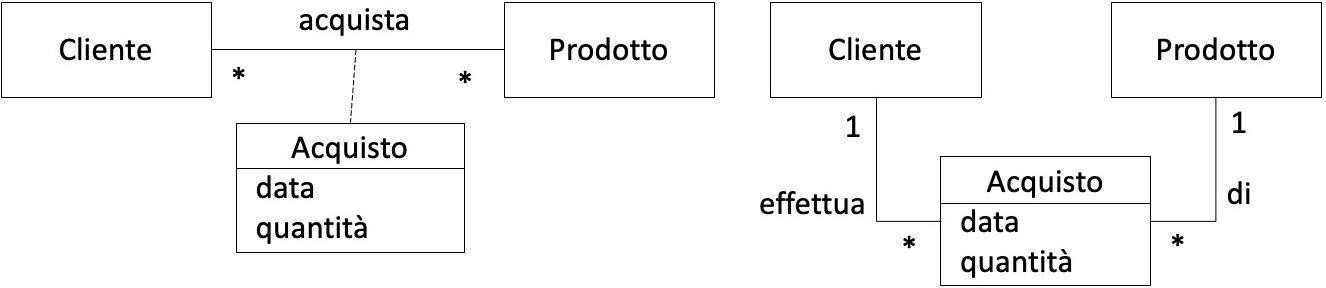
temperatura>=200 [pressione>=100 ]

1. SI
2. NO
3. solo quando o temperatura o pressione si mantengono costanti
4. In un diagramma di deployment, quali tra i seguenti possono essere modellati come *manufatti*?
5. un file sorgente Java
6. un file eseguibile
7. un PC usato come server
8. l’ambiente Apache
9. Quali sono i risultati attesi da uno *studio di fattibilità*?
10. stima dei costi del software
11. stima dei tempi di realizzazione del software
12. valutazione dei benefici del software
13. macroanalisi dei requisiti
14. progetto di massima del software
15. casi d’uso del software
16. A quali di questi criteri è consigliato attenersi nella *scelta dei colori* per un’interfaccia grafica?
17. basarsi su un codice di due soli colori
18. basarsi su un codice di non più di cinque colori
19. usare colori vivaci per aree grandi e neutri per aree piccole
20. non usare colori contrastanti tra loro per evitare affaticamento della vista
21. se lo sfondo è chiaro, usare un colore scuro per il testo
22. usare colori brillanti per applicazioni gestionali
23. Cosa afferma il *principio di anticipazione dei cambiamenti*?
24. un software deve soddisfare esclusivamente le specifiche funzionali attuali
25. un software deve soddisfare sia le specifiche funzionali attuali sia quelle prevedibili per il breve/medio termine
26. un software deve soddisfare le specifiche funzionali attuali ma essere predisposto per poter soddisfare, con bassi costi di manutenzione, anche quelle prevedibili per il breve/medio termine

# Compito di IS del 12/7/2021

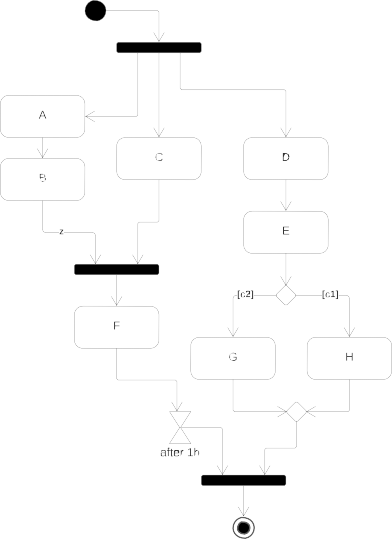
1. Si confrontino i due seguenti frammenti di *diagrammi delle classi*:



1. sono perfettamente equivalenti
2. quello di sinistra è più generale
3. quello di destra è più generale
4. Cosa si intende per *robustezza* di un software?
5. la sua capacità di reagire in modo "ragionevole" anche a fronte di situazioni non previste dalle specifiche
6. l'assenza di bug
7. il fatto che rispetta le specifiche di progetto
8. la sua capacità di cooperare con altri software
9. Quali tipi di relazioni tra package possono comparire in un *diagramma dei package* in UML 2?
10. contenimento
11. specializzazione
12. associazione
13. dipendenza
14. aggregazione
15. interazione
16. Quali tra le seguenti sequenze, risultanti dall'applicazione della tecnica di *analisi del flusso dei dati* a una variabile usata in un frammento di codice, vengono considerate anomale?
17. duuduua
18. aduuddua
19. dauuduua
20. aduuaduua
21. Quale dei seguenti *tipi di interfaccia* è da ritenersi ottimale per un'applicazione di acquisto biglietti da installare su un totem presso una stazione ferroviaria?
22. interfaccia code-based
23. interfaccia 3270
24. pseudo-GUI
25. standard GUI
26. special GUI
27. Quali delle seguenti affermazioni riguardanti la *prototipazione usa-e-getta* sono vere?
28. permette di dimostrare in anticipo i requisiti agli utenti
29. tende a corrompere il sistema rendendone costosa la manutenzione
30. si focalizza sui requisiti meno chiari
31. può essere usata per prototipare una interfaccia utente

# Compito di IS del 15/9/2021

1. In quale delle seguenti fasi del ciclo di vita vengono definite le *specifiche* per un software?
2. pianificazione
3. analisi
4. progettazione del sistema
5. progettazione esecutiva
6. realizzazione
7. collaudo
8. Un software viene modificato per cambiare la funzione di calcolo di una provvigione a seguito di una disposizione di legge. Di che tipo di *manutenzione* si tratta?
9. Correttiva
10. Evolutiva
11. Perfettiva
12. Adattiva
13. Nell’ambito di Unified Process (UP), cosa si intende per *elaboration*?
14. la fase in cui si definiscono gli obiettivi di progetto
15. la fase in cui vengono definite le caratteristiche funzionali, strutturali e architetturali del software
16. la fase in cui il software viene sviluppato e collaudato
17. la fase in cui il software viene consegnato, installato e configurato
18. Quali delle seguenti affermazioni generali relative al *linguaggio UML* sono vere?
19. UML è un linguaggio standard basato su un metamodello
20. UML deve essere utilizzato in abbinamento a Unified Process
21. un’entità di UML può essere rappresentata con notazioni diverse a seconda del diagramma in cui compare
22. un’entità di UML può essere rappresentata con notazioni diverse a seconda del livello di dettaglio che si vuole mostrare
23. un modello è una particolare visualizzazione di alcuni tipi di elementi di un diagramma
24. una associazione è un caso particolare di aggregazione
25. se una classe è un’interfaccia, allora è una classe astratta
26. se una classe è astratta, allora è una interfaccia
27. Dato il *diagramma di attività* in figura, quali delle seguenti affermazioni riguardanti le azioni raffigurate sono vere?



1. A deve essere svolta prima di C
2. B viene terminata dall’evento z
3. D può essere svolta prima di B
4. F può essere svolta prima di D
5. se la condizione c1 è falsa viene eseguita G
6. dopo un’ora dal termine di F, l’attività termina
7. F può iniziare solo quando C è terminata
8. G e H possono essere svolte concorrentemente
9. B e C possono essere svolte concorrentemente

**7)** E’ data una funzionalità di stampa dell’anagrafica completa di un utente. In quale modo può essere classificata nel metodo *function points*?

1. ILF
2. EIF
3. EI
4. EO
5. EQ