


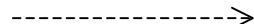

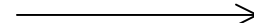
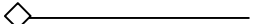

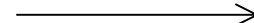
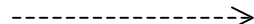

Ingegneria del Software (matr. pari-dispari): Esempi di domande a risposta multipla

1.	Cosa è un metodo ? <ul style="list-style-type: none"><input type="checkbox"/> Un sistema per fare qualcosa in modo migliore<input type="checkbox"/> Un procedimento generale per risolvere classi di problemi<input type="checkbox"/> Un particolare approccio o filosofia per fare qualcosa
2.	Quale di queste affermazioni non è appropriata per il modello di processo a cascata ? <ul style="list-style-type: none"><input type="checkbox"/> Le fasi del processo sono in progressione sequenziale<input type="checkbox"/> I semilavorati all'uscita di una fase sono congelati e non possono essere più modificati<input type="checkbox"/> Le funzionalità del sistema sono sviluppate in maniera incrementale
3.	Un prototipo viene sviluppato per: <ul style="list-style-type: none"><input type="checkbox"/> Ridurre i costi di sviluppo<input type="checkbox"/> Rilasciare rapidamente al cliente una prima versione del sistema<input type="checkbox"/> Interagire con il committente per convalidare i requisiti
4.	Quale tipo di prototipazione parte con i requisiti meglio compresi ? <ul style="list-style-type: none"><input type="checkbox"/> Prototipazione esplorativa<input type="checkbox"/> Prototipazione throw-away<input type="checkbox"/> Prototipazione mock-ups
5.	Quale tipo di prototipazione tende a realizzare l'interfaccia utente? <ul style="list-style-type: none"><input type="checkbox"/> Prototipazione esplorativa<input type="checkbox"/> Prototipazione breadboards<input type="checkbox"/> Prototipazione mock-ups
6.	Quale di queste affermazioni è appropriata per il modello di processo incrementale ? <ul style="list-style-type: none"><input type="checkbox"/> Ogni versione produce funzionalità/sottosistemi più affidabili<input type="checkbox"/> Ogni versione aggiunge nuove funzionalità/sottosistemi al sistema<input type="checkbox"/> Ogni versione raffina le funzionalità/sottosistemi che sono presenti fin dall'inizio
7.	Quale di queste affermazioni è appropriata per il modello di processo iterativo ? <ul style="list-style-type: none"><input type="checkbox"/> Ogni versione raffina le funzionalità/sottosistemi che sono presenti fin dall'inizio<input type="checkbox"/> Ogni versione aggiunge nuove funzionalità/sottosistemi al sistema<input type="checkbox"/> I requisiti a più alta priorità vengono rilasciati per primi

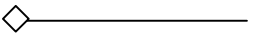

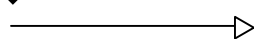
8.	Cosa è un team ?
	<input type="checkbox"/> Un insieme di persone a cui è assegnato un task comune, ma che lavorano individualmente e senza necessità di interazione <input type="checkbox"/> Un insieme di persone che rivedono un work product e che propongono azioni <input type="checkbox"/> Un piccolo insieme di persone che lavorano in stretta interazione sulla stessa attività o task
9.	Il tempo durante il quale un certo lavoro in un progetto deve essere fatto, viene denominato:
	<input type="checkbox"/> Work-product <input type="checkbox"/> Schedule <input type="checkbox"/> Task
10.	Il lavoro che deve essere eseguito da un partecipante al progetto, viene denominato:
	<input type="checkbox"/> Work-product <input type="checkbox"/> Schedule <input type="checkbox"/> Task
11.	Quale di queste affermazioni non è valida per una struttura organizzativa di reporting gerarchica ?
	<input type="checkbox"/> Lo stato è riportato dall'alto verso il basso <input type="checkbox"/> Le decisioni vengono comunicate dall'alto verso il basso <input type="checkbox"/> Le informazioni relative a stato e decisioni sono unidirezionali
12.	Cosa è un liason all'interno di un team ?
	<input type="checkbox"/> Il leader del team <input type="checkbox"/> Il responsabile della comunicazione con un altro team <input type="checkbox"/> Il consulente amministrativo del team
13.	Che tipo di ruolo è un configuration manager ?
	<input type="checkbox"/> Uno sviluppatore <input type="checkbox"/> Un manager <input type="checkbox"/> Un liason
14.	Che cosa è un GANNT ?
	<input type="checkbox"/> Un grafico a barre che descrive lo schedule e la durata dei vari task <input type="checkbox"/> Un grafo che descrive le dipendenze tra i task del progetto <input type="checkbox"/> Un grafo che descrive le dipendenze tra i work-product del progetto
15.	Che cosa è un PERT?
	<input type="checkbox"/> Un grafico a barre che descrive lo schedule e la durata dei vari task <input type="checkbox"/> Un grafo che descrive le dipendenze tra i task del progetto <input type="checkbox"/> Un grafo che descrive le dipendenze tra i work-product del progetto

16.	Quale dei seguenti non è una comunicazione pianificata?
	<input type="checkbox"/> Definizione del problema <input type="checkbox"/> Review di progetto <input type="checkbox"/> Richiesta di modifica
17.	Quale dei seguenti è una comunicazione non pianificata?
	<input type="checkbox"/> Risoluzione di un problema <input type="checkbox"/> Ispezione <input type="checkbox"/> Rilascio
18.	Quale dei seguenti meccanismi di comunicazione è asincrono?
	<input type="checkbox"/> Intervista strutturata <input type="checkbox"/> Riunione <input type="checkbox"/> Posta elettronica
19.	Cosa è un modello ?
	<input type="checkbox"/> Un'astrazione che descrive il sistema o un sottoinsieme di un sistema <input type="checkbox"/> Un insieme di regole grafiche o testuali per rappresentare viste <input type="checkbox"/> Una vista di aspetti del sistema
20.	Cosa è una vista ?
	<input type="checkbox"/> Un'astrazione che descrive il sistema o un sottoinsieme di un sistema <input type="checkbox"/> Una visualizzazione di particolari aspetti di un modello <input type="checkbox"/> Un insieme di regole grafiche o testuali
21.	Cosa è una notazione ?
	<input type="checkbox"/> Un'astrazione che descrive il sistema o un sottoinsieme di un sistema <input type="checkbox"/> Una visualizzazione di particolari aspetti di un modello <input type="checkbox"/> Un insieme di regole grafiche o testuali per rappresentare viste
22.	Cosa è UML ?
	<input type="checkbox"/> Una notazione grafica per progettare sistemi software <input type="checkbox"/> Un insieme di linguaggi per modellare software <input type="checkbox"/> Un modello astratto per descrivere sistemi software
23.	Quale di questi diagrammi non è usato per descrivere il comportamento dinamico di un sistema software ?
	<input type="checkbox"/> Sequence diagram <input type="checkbox"/> Activity diagram <input type="checkbox"/> Use case diagram

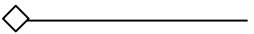
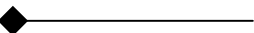
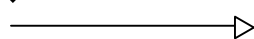
24.	<p>I diagrammi dei casi d'uso:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Descrivono il comportamento funzionale del sistema così come visto dagli utenti <input type="checkbox"/> Descrivono il comportamento dinamico tra gli attori e il sistema e tra gli oggetti del sistema <input type="checkbox"/> Descrivono il comportamento dinamico di un sistema, in particolare il workflow
25.	<p>I diagrammi delle attività:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Descrivono il comportamento funzionale del sistema così come visto dagli utenti <input type="checkbox"/> Descrivono il comportamento dinamico tra gli attori e il sistema e tra gli oggetti del sistema <input type="checkbox"/> Descrivono il comportamento dinamico di un sistema, in particolare il workflow
26.	<p>I diagrammi di sequenza:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Descrivono il comportamento funzionale del sistema così come visto dagli utenti <input type="checkbox"/> Descrivono il comportamento dinamico tra gli attori e il sistema e tra gli oggetti del sistema <input type="checkbox"/> Descrivono il comportamento dinamico di un sistema, in particolare il workflow
27.	<p>I diagrammi di stato:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Descrivono la struttura statica del sistema: oggetti, attributi e relazioni <input type="checkbox"/> Descrivono il comportamento dinamico tra gli attori e il sistema e tra gli oggetti del sistema <input type="checkbox"/> Descrivono il comportamento dinamico di un singolo oggetto come una macchina a stati finiti
28.	<p>I diagrammi delle classi:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Descrivono la struttura statica del sistema: oggetti, attributi e relazioni <input type="checkbox"/> Descrivono il comportamento dinamico tra gli attori e il sistema e tra gli oggetti del sistema <input type="checkbox"/> Descrivono il comportamento dinamico di un singolo oggetto come una macchina a stati finiti
29.	<p>In UML le classi e gli oggetti (istanze) sono rappresentate con:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Rettangoli <input type="checkbox"/> Ovali <input type="checkbox"/> Rettangoli con angoli arrotondati
30.	<p>In UML i casi d'uso sono rappresentati con:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Rettangoli <input type="checkbox"/> Ovali <input type="checkbox"/> Rettangoli con angoli arrotondati
31.	<p>In UML gli stati sono rappresentati con:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Rettangoli <input type="checkbox"/> Ovali <input type="checkbox"/> Rettangoli con angoli arrotondati

<p>32. Quale di questi elementi non è parte di un diagramma dei casi d'uso:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Attore <input type="checkbox"/> Caso d'uso <input type="checkbox"/> Classe
<p>33. Quale di questi elementi non è parte di un diagramma di sequenza:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Attore <input type="checkbox"/> Caso d'uso <input type="checkbox"/> Oggetti
<p>34. Che tipo di relazione può esistere tra un attore e un caso d'uso:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Associazione <input type="checkbox"/> Dipendenza <input type="checkbox"/> Generalizzazione
<p>35. Che tipo di relazione può esistere tra due attori:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Associazione <input type="checkbox"/> Dipendenza <input type="checkbox"/> Generalizzazione
<p>36. Quale di queste relazioni non esiste tra due casi d'uso:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Associazione <input type="checkbox"/> Dipendenza <input type="checkbox"/> Generalizzazione
<p>37. Quali di queste relazioni rappresenta una dipendenza:</p> <ul style="list-style-type: none"> <input type="checkbox"/>  <input type="checkbox"/>  <input type="checkbox"/> 
<p>38. Quali di queste relazioni rappresenta una generalizzazione:</p> <ul style="list-style-type: none"> <input type="checkbox"/>  <input type="checkbox"/>  <input type="checkbox"/> 
<p>39. Quali di queste relazioni rappresenta una associazione:</p> <ul style="list-style-type: none"> <input type="checkbox"/>  <input type="checkbox"/>  <input type="checkbox"/> 

40. Quali di queste relazioni rappresenta una aggregazione:

- ☐  _____
- ☐  _____
- ☐ _____ 

41. Quali di queste relazioni rappresenta una composizione:

- ☐  _____
- ☐  _____
- ☐ _____ 

42. Quali di questi è un requisito funzionale:

- ☐ Il sistema deve visualizzare l'ora in base alla sua locazione
- ☐ Il tempo di risposta deve essere meno di un secondo
- ☐ Il linguaggio di implementazione deve essere Java

43. Quali di questi è un requisito non funzionale:

- ☐ Il sistema deve visualizzare l'ora in base alla sua locazione
- ☐ Il tempo di risposta deve essere meno di un secondo
- ☐ Il linguaggio di implementazione deve essere Java

44. Cosa si intende per correttezza dei requisiti ?

- ☐ I requisiti rappresentano la vista dell'utente
- ☐ Sono descritti tutti i possibili scenari del sistema
- ☐ Non ci sono requisiti funzionali e non funzionali che si contraddicono

45. Cosa si intende per completezza dei requisiti ?

- ☐ I requisiti rappresentano la vista dell'utente
- ☐ Sono descritti tutti i possibili scenari del sistema
- ☐ Non ci sono requisiti funzionali e non funzionali che si contraddicono

46. Cosa si intende per consistenza dei requisiti ?

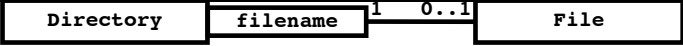
- ☐ I requisiti rappresentano la vista dell'utente
- ☐ Sono descritti tutti i possibili scenari del sistema
- ☐ Non ci sono requisiti funzionali e non funzionali che si contraddicono

47. Cosa si intende per chiarezza dei requisiti ?

- ☐ Non ci sono ambiguità nei requisiti
- ☐ I requisiti possono essere implementati e rilasciati
- ☐ Non ci sono requisiti funzionali e non funzionali che si contraddicono

48.	Cosa si intende per realismo dei requisiti ?
	<input type="checkbox"/> Non ci sono ambiguità nei requisiti <input type="checkbox"/> I requisiti possono essere implementati e rilasciati <input type="checkbox"/> Per ogni funzione del sistema è possibile individuare un insieme di requisiti funzionali
49.	Cosa si intende per tracciabilità dei requisiti ?
	<input type="checkbox"/> Non ci sono ambiguità nei requisiti <input type="checkbox"/> I requisiti possono essere implementati e rilasciati <input type="checkbox"/> Per ogni funzione del sistema è possibile individuare un insieme di requisiti funzionali
50.	Quale di questi requisiti vanno affrontati durante il design ma non durante l'implementazione ?
	<input type="checkbox"/> Requisiti ad alta priorità <input type="checkbox"/> Requisiti a media priorità <input type="checkbox"/> Requisiti a bassa priorità
51.	Cosa si intende per interface engineering ?
	<input type="checkbox"/> Lo sviluppo parte da zero, non esiste un sistema precedente <input type="checkbox"/> Riprogettazione o reimplementazione di un sistema esistente con nuove tecnologie <input type="checkbox"/> Fornire i servizi di un sistema esistente in un nuovo ambiente operativo
52.	Che tipo di scenari sono usati per descrivere un sistema futuro?
	<input type="checkbox"/> as-is scenarios <input type="checkbox"/> visionary scenarios <input type="checkbox"/> training scenarios
53.	Che tipo di scenari sono usati per descrivere un sistema esistente ?
	<input type="checkbox"/> as-is scenarios <input type="checkbox"/> visionary scenarios <input type="checkbox"/> training scenarios
54.	Che tipo di scenari sono usati per guidare un utente nuovo nell'utilizzo del sistema ?
	<input type="checkbox"/> as-is scenarios <input type="checkbox"/> visionary scenarios <input type="checkbox"/> training scenarios
55.	Perché si usano gli scenari ?
	<input type="checkbox"/> Per individuare i requisiti del sistema <input type="checkbox"/> Per analizzare le associazioni tra gli oggetti del sistema <input type="checkbox"/> Per analizzare il flusso di eventi di un caso d'uso

56.	Come si descrive un caso d'uso ?
	<input type="checkbox"/> Mediante uno scenario <input type="checkbox"/> Mediante un flusso di eventi <input type="checkbox"/> Mediante un insieme di requisiti funzionali
57.	Cosa esprime una relazione di inclusione tra casi d'uso ?
	<input type="checkbox"/> Una variante del normale flusso di eventi di un caso d'uso <input type="checkbox"/> Una specializzazione di un caso d'uso <input type="checkbox"/> Una decomposizione funzionale di un caso d'uso
58.	Cosa esprime una relazione di estensione tra casi d'uso ?
	<input type="checkbox"/> Una variante del normale flusso di eventi di un caso d'uso <input type="checkbox"/> Una specializzazione di un caso d'uso <input type="checkbox"/> Una decomposizione funzionale di un caso d'uso
59.	Cosa esprime una relazione di generalizzazione tra casi d'uso ?
	<input type="checkbox"/> Una variante del normale flusso di eventi di un caso d'uso <input type="checkbox"/> Una specializzazione di un caso d'uso <input type="checkbox"/> Una decomposizione funzionale di un caso d'uso
60.	Un oggetto che interagisce con un attore in un sequence diagram è:
	<input type="checkbox"/> Un entity object <input type="checkbox"/> Un boundary object <input type="checkbox"/> Un control object
61.	Quale di queste categorie di requisiti non funzionali indica la facilità di cambiamenti al sistema dopo il rilascio ?
	<input type="checkbox"/> Reliability <input type="checkbox"/> Performance <input type="checkbox"/> Supportability
62.	Quale di queste categorie di pseudo-requirements indica vincoli legati alle modalità di rilascio del sistema?
	<input type="checkbox"/> Interface requirements <input type="checkbox"/> Packaging requirements <input type="checkbox"/> Operations requirements
63.	Una trasformazione applicata agli oggetti di una classe e definita in fase di analisi è chiamata:
	<input type="checkbox"/> Operazione <input type="checkbox"/> Signature <input type="checkbox"/> Metodo

64.	Una connessione tra due istanze di oggetti è chiamata:
	<input type="checkbox"/> Associazione <input type="checkbox"/> Link <input type="checkbox"/> Relazione
65.	Una associazione è:
	<input type="checkbox"/> Una connessione tra due istanze di oggetti <input type="checkbox"/> Una dipendenza tra classi <input type="checkbox"/> Un mapping bidirezionale tra classi
66.	Quale di questi tipi di oggetti cambia meno frequentemente in un sistema software ?
	<input type="checkbox"/> Boundary object <input type="checkbox"/> Control object <input type="checkbox"/> Entity object
67.	Cosa indica un ruolo ?
	<input type="checkbox"/> Una estremità dell'associazione <input type="checkbox"/> La navigabilità dell'associazione <input type="checkbox"/> La direzione del nome dell'associazione
68.	Cosa indica la direzione di una associazione ?
	<input type="checkbox"/> Una estremità dell'associazione <input type="checkbox"/> La navigabilità dell'associazione <input type="checkbox"/> La direzione del nome dell'associazione
69.	Nel diagramma seguente, "filename" 
	<input type="checkbox"/> E' un attributo della classe directory <input type="checkbox"/> E' un attributo della classe file <input type="checkbox"/> E' il nome della associazione
70.	Cosa indicano le frecce in un diagramma di sequenza ?
	<input type="checkbox"/> Eventi inviati da un oggetto ad un altro <input type="checkbox"/> Relazioni tra oggetti <input type="checkbox"/> Dipendenze tra oggetti
71.	Cosa indica la linea tratteggiata verticale sotto un oggetto in un diagramma di sequenza ?
	<input type="checkbox"/> L'invio di un messaggio all'oggetto <input type="checkbox"/> Il periodo di attivazione di una operazione dell'oggetto <input type="checkbox"/> Il periodo durante il quale è possibile inviare un messaggio all'oggetto

72.	Cosa indica un rettangolo verticale sotto un oggetto in un diagramma di sequenza ?
	<input type="checkbox"/> L'invio di un messaggio all'oggetto <input type="checkbox"/> Il periodo di attivazione di una operazione dell'oggetto <input type="checkbox"/> Il periodo durante il quale è possibile inviare un messaggio all'oggetto
73.	Cosa dovrebbe contenere la prima colonna di un diagramma di sequenza ?
	<input type="checkbox"/> L'attore che inizia il caso d'uso <input type="checkbox"/> Un boundary object <input type="checkbox"/> Il control object che gestisce il caso d'uso
74.	Cosa dovrebbe contenere la seconda colonna di un diagramma di sequenza ?
	<input type="checkbox"/> L'attore che inizia il caso d'uso <input type="checkbox"/> Un boundary object <input type="checkbox"/> Il control object che gestisce il caso d'uso
75.	Cosa dovrebbe contenere la terza colonna di un diagramma di sequenza ?
	<input type="checkbox"/> L'attore che inizia il caso d'uso <input type="checkbox"/> Un boundary object <input type="checkbox"/> Il control object che gestisce il caso d'uso
76.	Quali di queste affermazioni non è corretta ?
	<input type="checkbox"/> Un boundary object accede ad un entity object <input type="checkbox"/> Un entity object accede ad un control object <input type="checkbox"/> Un control object crea un boundary object
77.	In un diagramma di stato, quando viene valutata la condizione di guardia su una transizione ?
	<input type="checkbox"/> Quando il sistema si trova nello stato sorgente della transizione <input type="checkbox"/> Quando si verifica l'azione indicata sulla transizione <input type="checkbox"/> Quando si verifica l'evento indicato sulla transizione
78.	Quale di queste affermazioni non si riferisce ad un'attività in un diagramma di stato ?
	<input type="checkbox"/> Operazione che prende tempo per essere completata <input type="checkbox"/> E' associata ad un evento <input type="checkbox"/> E' associata ad uno stato
79.	Quale di queste affermazioni non si riferisce ad un'azione in un diagramma di stato ?
	<input type="checkbox"/> Operazione che prende tempo per essere completata <input type="checkbox"/> E' associata ad uno stato <input type="checkbox"/> E' associata ad un evento

80.	Quale di queste affermazioni si riferisce ad un diagramma di stato ?
	<input type="checkbox"/> Serve ad individuare le relazioni temporali tra oggetti nel tempo <input type="checkbox"/> Serve ad individuare la sequenza di operazioni come risposta ad uno o più eventi <input type="checkbox"/> Serve ad individuare i cambiamenti degli oggetti nel tempo
81.	Quale di queste affermazioni non si riferisce ad un diagramma di sequenza ?
	<input type="checkbox"/> Serve ad individuare i cambiamenti degli oggetti nel tempo <input type="checkbox"/> Serve ad individuare le relazioni temporali tra oggetti nel tempo <input type="checkbox"/> Serve ad individuare la sequenza di operazioni come risposta ad uno o più eventi
82.	Quale di questi prodotti dell'analisi dei requisiti vanno usati durante la fase di definizione degli obiettivi di design ?
	<input type="checkbox"/> Modello dei casi d'uso <input type="checkbox"/> Modello a oggetti <input type="checkbox"/> Requisiti non funzionali
83.	Quale di questi prodotti dell'analisi dei requisiti vanno usati durante la fase di decomposizione del sistema ?
	<input type="checkbox"/> Modello dei casi d'uso <input type="checkbox"/> Modello a oggetti <input type="checkbox"/> Requisiti non funzionali
84.	Quale di questi prodotti dell'analisi dei requisiti vanno usati durante la fase di design relativa a mapping hardware/software e a gestione dei dati persistenti ?
	<input type="checkbox"/> Modello dei casi d'uso <input type="checkbox"/> Modello a oggetti <input type="checkbox"/> Requisiti non funzionali
85.	Come viene rappresentato un sottosistema in UML ?
	<input type="checkbox"/> Con una classe <input type="checkbox"/> Con un caso d'uso <input type="checkbox"/> Con un package
86.	Cosa è un servizio di un sottosistema ?
	<input type="checkbox"/> Un insieme di operazioni con signature completamente specificata <input type="checkbox"/> Un gruppo di operazioni che condividono uno scopo comune <input type="checkbox"/> Un insieme di associazioni, eventi e vincoli legati tra di loro
87.	Cosa è l'interfaccia di un sottosistema ?
	<input type="checkbox"/> Un insieme di operazioni con signature completamente specificata <input type="checkbox"/> Un gruppo di operazioni che condividono uno scopo comune <input type="checkbox"/> Un insieme di associazioni, eventi e vincoli legati tra di loro

<p>88. Quali di queste affermazioni è vera ?</p> <ul style="list-style-type: none"> <input type="checkbox"/> L'accoppiamento misura le dipendenze tra le classi di un sottosistema <input type="checkbox"/> In un sistema con elevato accoppiamento le modifiche ad un sottosistema hanno forte impatto sugli altri sottosistemi <input type="checkbox"/> L'obiettivo del system design è massimizzare l'accoppiamento
<p>89. Quali di queste affermazioni è vera ?</p> <ul style="list-style-type: none"> <input type="checkbox"/> La coesione misura le dipendenze tra i sottosistemi di un sistema <input type="checkbox"/> In sottosistemi con elevata coesione le modifiche ad un sottosistema hanno forte impatto sugli altri sottosistemi <input type="checkbox"/> Le classi di un sottosistema con elevata coesione eseguono task simili
<p>90. A cosa servono le partizioni di un sistema ?</p> <ul style="list-style-type: none"> <input type="checkbox"/> A dividere verticalmente un sistema in sottosistemi debolmente accoppiati <input type="checkbox"/> A dividere orizzontalmente un sistema in sottosistemi a diversi livelli di astrazione <input type="checkbox"/> A dividere un sistema sia orizzontalmente che verticalmente in sottosistemi indipendenti
<p>91. A cosa servono i layer di un sistema ?</p> <ul style="list-style-type: none"> <input type="checkbox"/> A dividere verticalmente un sistema in sottosistemi debolmente accoppiati <input type="checkbox"/> A dividere orizzontalmente un sistema in sottosistemi a diversi livelli di astrazione <input type="checkbox"/> A dividere un sistema sia orizzontalmente che verticalmente in sottosistemi indipendenti
<p>92. A chi fornisce servizi un layer di un sistema ?</p>
<ul style="list-style-type: none"> <input type="checkbox"/> Ai layer di livello più basso
<ul style="list-style-type: none"> <input type="checkbox"/> Ai layer dello stesso livello
<ul style="list-style-type: none"> <input type="checkbox"/> Ai layer di livello più alto
<p>93. In un'architettura software aperta:</p> <ul style="list-style-type: none"> <input type="checkbox"/> I sottosistemi di un livello possono accedere solo ai sottosistemi del livello immediatamente inferiore <input type="checkbox"/> I sottosistemi di un livello possono accedere ai sottosistemi di qualunque livello inferiore <input type="checkbox"/> I sottosistemi di un livello possono accedere sia ai sottosistemi dei livelli inferiori che a quelli dei livelli superiori
<p>94. In un'architettura software chiusa:</p> <ul style="list-style-type: none"> <input type="checkbox"/> I sottosistemi di un livello possono accedere solo ai sottosistemi del livello immediatamente inferiore <input type="checkbox"/> I sottosistemi di un livello possono accedere solo ai sottosistemi dello stesso livello <input type="checkbox"/> I sottosistemi di un livello possono accedere ai sottosistemi di qualunque livello inferiore

95.	Quali sono gli obiettivi di design di un'architettura software aperta ?
	<input type="checkbox"/> Manutenibilità <input type="checkbox"/> Efficienza <input type="checkbox"/> Affidabilità
96.	Quali sono gli obiettivi di design di un'architettura software chiusa ?
	<input type="checkbox"/> Usabilità <input type="checkbox"/> Efficienza <input type="checkbox"/> Manutenibilità
97.	Quale di queste affermazioni è falsa?
	<input type="checkbox"/> In un'architettura client-server, il server fornisce i servizi al client <input type="checkbox"/> In un'architettura client-server, il server conosce l'interfaccia del client <input type="checkbox"/> Una repository architecture è un caso particolare di architettura client-server
98.	Quali di queste affermazioni è vera ?
	<input type="checkbox"/> In un'architettura peer-to-peer non c'è differenza tra client e server <input type="checkbox"/> Un'architettura peer-to-peer favorisce una gestione centralizzata dei dati <input type="checkbox"/> Un'architettura peer-to-peer consente di evitare problemi di deadlock
99.	Un'architettura model-view-controller, il sottosistema "model":
	<input type="checkbox"/> E' responsabile della conoscenza del dominio applicativo <input type="checkbox"/> E' responsabile della visualizzazione degli oggetti del dominio applicativo <input type="checkbox"/> E' responsabile della sequenza dei interazioni con l'utente
100.	Dire quali di questi diagrammi model-view-controller è corretto:
<input type="checkbox"/>	<pre> classDiagram class Controller class Model class View class Repository Controller "*" -- "1" Model : initiator Model "1" -- "*" View : subscriber Model "1" -- "1" Repository : repository </pre>
<input type="checkbox"/>	<pre> classDiagram class Controller class Model class View class Repository Controller "*" -- "1" Repository : initiator Model "1" -- "*" View : subscriber Model "1" -- "1" Repository : repository </pre>
<input type="checkbox"/>	<pre> classDiagram class Controller class Model class View class Repository Controller "*" -- "1" Repository : initiator Model "1" -- "*" View : notifier Model "1" -- "1" Repository : subscriber </pre>

101.	Le relazioni tra i componenti in un component diagram sono
	<input type="checkbox"/> associazioni <input type="checkbox"/> dipendenze <input type="checkbox"/> generalizzazioni
102.	Quale di queste affermazioni relative ad un component diagram è falsa ?
	<input type="checkbox"/> Mostra il mapping hardware/software <input type="checkbox"/> Mostra la struttura del sistema a compilation time <input type="checkbox"/> Mostra la struttura a design time
103.	Quale di queste affermazioni relative ad un deployment diagram è falsa ?
	<input type="checkbox"/> Mostra la struttura del sistema a run-time <input type="checkbox"/> Mostra il mapping hardware/software <input type="checkbox"/> Mostra le dipendenze tra componenti e interfacce dei sottosistemi
104.	Le relazioni tra i nodi di un deployment diagram sono:
	<input type="checkbox"/> associazioni <input type="checkbox"/> dipendenze <input type="checkbox"/> generalizzazioni
105.	In un thin client model
	<input type="checkbox"/> Il sistema è two thier e la logica applicativa è eseguita sul server <input type="checkbox"/> Il sistema è two thier e la logica applicativa è eseguita sul client <input type="checkbox"/> Il sistema è three thier e la logica applicativa è eseguita sul client
106.	Se il mio obiettivo è fornire implementazioni diverse per uno stesso sottosistema
	<input type="checkbox"/> Uso un bridge pattern <input type="checkbox"/> Uso un adapter pattern <input type="checkbox"/> Uso un façade pattern
107.	Se il mio obiettivo è quello di realizzare una architettura chiusa
	<input type="checkbox"/> Uso un adapter pattern <input type="checkbox"/> Uso un bridge pattern <input type="checkbox"/> Uso un façade pattern
108.	Un invariante è
	<input type="checkbox"/> Un predicato che deve essere vero prima dell'invocazione di un metodo di una classe <input type="checkbox"/> Un predicato che è vero dopo l'invocazione di un metodo di una classe <input type="checkbox"/> Un predicato che è vero prima e dopo l'invocazione di un metodo di una classe

109. Quale di queste affermazioni relative a JavaDoc è corretta ?
<input type="checkbox"/> Il commento che precede un metodo consente di specificare la preconditione del metodo <input type="checkbox"/> Il commento che precede un metodo consente di specificare la postcondizione del metodo <input type="checkbox"/> Il commento che precede un metodo consente di specificare l'invariante della classe
110. Se devo realizzare una associazione qualificata uso come struttura dati
<input type="checkbox"/> Una tabella <input type="checkbox"/> Un insieme <input type="checkbox"/> Una lista
111. Se devo estendere una classe mi interessa conoscere
<input type="checkbox"/> Tutti i membri privati, protetti e pubblici della classe <input type="checkbox"/> Solo i membri protetti e pubblici della classe <input type="checkbox"/> Solo i membri pubblici della classe
112. La realizzazione delle associazioni del modello a oggetti
<input type="checkbox"/> E' una trasformazione del modello a oggetti <input type="checkbox"/> E' una trasformazione di tipo forward engineering <input type="checkbox"/> E' una trasformazione di refactoring
113. Se devo memorizzare dati voluminosi e necessari per un periodo breve di tempo
<input type="checkbox"/> uso una struttura dati in memoria <input type="checkbox"/> uso un file <input type="checkbox"/> uso un database
114. Se devo effettuare una trasformazione prima sul codice sorgente e poi sul modello a oggetti faccio
<input type="checkbox"/> Prima operazioni di forward engineering e poi di refactoring <input type="checkbox"/> Prima operazioni di object model transformation e poi di forward engineering <input type="checkbox"/> Prima operazioni di refactoring e poi di reverse engineering
115. Se eredito un contratto in UML quale di queste affermazioni non è corretta ?
<input type="checkbox"/> La preconditione del metodo nella sottoclasse consente al metodo di gestire meno casi del corrispondente metodo nella superclasse <input type="checkbox"/> Il metodo nella sottoclasse deve assicurare la stessa postcondizione del corrispondente metodo nella superclasse <input type="checkbox"/> L'invariante della sottoclasse può essere più restrittivo dell'invariante della superclasse
116. Quali di questi operazioni non è opportuno implementare durante la realizzazione di contratti ?
<input type="checkbox"/> Controllare che la preconditione all'inizio di un metodo sia soddisfatta <input type="checkbox"/> Controllare che la postcondizione all'uscita di un metodo sia soddisfatta <input type="checkbox"/> Incapsulare il codice di controllo in metodi separati ai fini del riuso

117.	L'evento percepito dall'utente come differenza tra comportamento atteso di un sistema software e comportamento esibito dal sistema software viene denominato
	<input type="checkbox"/> Failure <input type="checkbox"/> Fault <input type="checkbox"/> Error
118.	Nel testing di integrazione bottom-up ho bisogno di realizzare
	<input type="checkbox"/> Test Driver <input type="checkbox"/> Test Stub <input type="checkbox"/> Sia test driver che test stub
119.	Quale di queste affermazioni è falsa ?
	<input type="checkbox"/> Nel testing white box la derivazione dei casi di test mira a coprire la maggior parte di una classe di elementi della struttura del codice <input type="checkbox"/> Nel testing white box il numero di casi di test da eseguire dipende dal criterio di copertura adottato <input type="checkbox"/> Nel testing white-box non si controlla se l'output prodotto in corrispondenza di un input è uguale all'output atteso
120.	Quando si effettua il testing di regressione ?
	<input type="checkbox"/> Dopo aver effettuato il test di integrazione e prima di effettuare il test di sistema <input type="checkbox"/> Dopo aver effettuato il testing di sistema e prima di effettuare il testing di accettazione <input type="checkbox"/> Dopo aver effettuato delle modifiche al codice per correggere un fault