

EXAMEN TEORIA D'ENGINYERIA DEL SOFTWARE 1

CURS 2006/2007

2^a Convocatòria

Tipus A

Cognoms i nom	Grup
---------------	------

TEST (7 punts)

ATENCIÓ: Preguntes ben contestades: +1

Preguntes mal contestades: -0.25

Preguntes no contestades: 0

1) Definició d'enginyeria del software:

- a) El conjunt format per programes d'ordinador, estructures de dades i documentació associada.
- b) És un llenguatge visual de modelat de sistemes de software.
- c) És la ciència que té per objectiu aconseguir la màxima cohesió i el mínim acoblament en la divisió modular d'un programa.
- d) L'estudi dels processos, mètodes i eines que permeten crear, gestionar i mantenir programari d'ordinador.
- e) Cap de les anteriors.

2) Un paradigma lineal - seqüencial de desenvolupament de software:

- a) Obliga a finalitzar cada activitat (anàlisi, disseny, codificació i prova) abans de començar la següent.
- b) Construeix el software de manera incremental resolent progressivament requeriments de la llista de control del projecte.
- c) És una variant del model prototipat que permet codificar abans de fer l'anàlisi de requeriments.
- d) És un model de desenvolupament de software lligat a la programació orientada a objecte.
- e) Cap de les anteriors.

3) Un requeriment no funcional:

- a) És el pseudocodi d'una funció o procediment del programa.
- b) Expressa el comportament desitjat del software.
- c) És una restricció imposada sobre la funcionalitat del software que en limita el seu disseny.
- d) Es descriu amb un diagrama de classes.
- e) Cap de les anteriors.

4) De quin tipus és el següent requeriment: "La compra via web de la tenda virtual DISCON ha de poder suportar 100 sessions simultànies?"

- a) Requeriment funcional.
- b) Requeriment de rendiment.
- c) Objectiu de disseny.
- d) Decisió de disseny.
- e) Cap de les anteriors.

5) De quin tipus és el següent requeriment. L'encarregat ha de poder consultar una la llista de comandes pendents i a continuació seleccionarà una comanda perquè s'imprimeixi?

- a) Requeriment funcional.
- b) Requeriment d'anàlisi.
- c) Requeriment no funcional.
- d) Decisió de disseny.
- e) Cap de les anteriors.

6) Quina de les següents afirmacions és certa:

- a) El disseny arquitectònic defineix la relació entre els principals elements estructurals del programa.
- b) El disseny arquitectònic transforma el model de dades de l'anàlisi en les estructures de dades que després s'implementaran.
- c) El disseny arquitectònic defineix la millor jerarquia de mòduls per aconseguir l'equilibri entre cohesió i acoblament.
- d) a) i c)
- e) Cap de les anteriors.

7) Quin nivell de cohesió té el mòdul que agrupa totes les funcions de l'actor encarregat: *preparar comanda, gestió estat comanda, reassignar horari repartiment comanda.*

- a) Funcional.
- b) Seqüencial.
- c) Comunicacional.
- d) Lògic.
- e) Cap de les anteriors.

- 8) Quina d'aquestes afirmacions sobre l'acoblament és certa:
- a) L'acoblament mesura el cost d'interacció entre els mòduls, és a dir, el cost relacionat amb la programació de les interfícies.
 - b) L'acoblament mesura la unicitat dels mòduls, és a dir que cada mòdul es dediqui a una única tasca.
 - c) L'acoblament mesura el número de mòduls que s'obtenen quan es passen al disseny els casos d'ús.
 - d) L'acoblament mesura el cost acumulat de programar un mòdul i el cost de programar la integració amb els altres.
 - e) Cap de les anteriors.
- 9) Quina és la diferència entre classe i objecte:
- a) Una classe és un mòdul i un objecte és una component (mòdul compilat).
 - b) Una classe i un objecte són el mateix, s'utilitzen els dos conceptes en funció de la metodologia orientada a objecte.
 - c) Una classe és la definició d'un tipus i objecte és una instanciació d'una classe.
 - d) Una classe és un agregat d'un objecte.
 - e) Cap de les anteriors.
- 10) En orientat a objecte es pot distingir els següents tipus de jerarquia:
- a) Jerarquia de processos i jerarquia de dades.
 - b) Jerarquia de generalització i jerarquia d'herència.
 - c) Jerarquia d'agregació/composició i jerarquia de generalització.
 - d) Jerarquia de tot/part i jerarquia de continent/contingut.
 - e) Cap de les anteriors.
- 11) En UML un cas d'ús és:
- a) Una entitat externa que interactua amb el software.
 - b) Un requeriment no funcional que afectarà a les decisions de disseny.
 - c) Una classe en l'etapa d'anàlisi.
 - d) Un requeriment funcional, és a dir la identificació d'una de les funcionalitats.
 - e) Cap de les anteriors.
- 12) En un diagrama de classes:
- a) Les relacions d'agregació representen una especialització d'una classe A respecte una classe B.
 - b) Les relacions d'agregació representen una no jerarquia entre dues classes.
 - c) Les relacions d'agregació representen una relació de tot-part d'un objecte A respecte un objecte B.
 - d) Les relacions d'agregació representen un missatge classe A respecte una classe B.
 - e) Cap de les anteriors.
- 13) Diferència entre agregació i composició:
- a) Una agregació té herència de propietats i una composició no en té.
 - b) Una composició és una agregació on les parts no poden sobreviure al tot.
 - c) No hi ha cap diferència, ambdues són sinònims d'associació entre classes.
 - d) Una agregació relaciona les classes d'anàlisi i una composició, les de disseny.
 - e) Cap de les anteriors.
- 14) En RUP, quin d'aquests artefactes es té en acabar la fase d'inici:
- a) Document de visió.
 - b) Document d'arquitectura.
 - c) Versió beta del software.
 - d) Producte final.
 - e) Cap de les anteriors.
- 15) En quin paradigma desenvolupament de software es pot classificar RUP:
- a) Lineal-seqüencial.
 - b) Prototipat.
 - c) Evolutiu.
 - d) Liberal.
 - e) Cap de les anteriors.
- 16) L'estructura estàtica de RUP es divideix en:
- a) Qui, què, com i quan.
 - b) Inici, elaboració, construcció i transició.
 - c) Model de negoci, model de requeriments, model d'anàlisi i disseny, model d'implementació, model de prova i model de desplegament.
 - d) El RUP no té estructura dinàmica, només estructura estàtica.
 - e) Cap de les anteriors.

- 17) En UML, en el diagrama de casos d'ús la <<extensió>> es defineix com:
- Un CU hereta el comportament i significat de l'altre
 - Un CU base incorpora explícitament el comportament d'un altre en algun punt de la seva seqüència.
 - Un CU base incorpora implícitament el comportament d'un altre CU en el lloc especificat indirectament per aquest altre CU. Un CU completa la funcionalitat d'un altre CU.
 - Un CU no pot incloure altres casos d'ús.
 - Cap de les anteriors.
- 18) En UML la vista de comportament ens dóna:
- Funcionalitats del sistema des del punt de vista de l'usuari.
 - Disseny de classes per implementar els requeriments funcionals i no funcionals.
 - Aspectes dinàmics del SW en execució: flux de missatges entre objectes, flux de control en una funció, canvis d'estat.
 - Organització del SW a l'entorn de desenvolupament.
 - Cap de les anteriors.
- 19) La classe d'anàlisi de Comunicació (Boundary):
- Gestionen la interacció entre el sistema i el seu entorn. S'utilitzen per modelar les interfícies del sistema.
 - Coordinen els events necessaris per realitzar el comportament especificat en un cas d'ús.
 - Modelen informació i el seu comportament. Representen entitats del món real o entitats internes necessàries per executar les tasques del sistema. Són independents de com l'entorn es comunica amb el sistema i sovint també independents de l'aplicació (transportables a altres aplicacions).
 - Utilitzem els punts de decisió per encaminar el flux d'events.
 - Cap de les anteriors.
- 20) La multiplicitat d'una relació d'associació defineix:
- El número de fluxos del cas d'ús.
 - Quants missatges es passen dues classes.
 - Quantes classes participen en la relació.
 - Quants objectes participen en la relació.
 - Cap de les anteriors.

RESPOSTES AL TEST

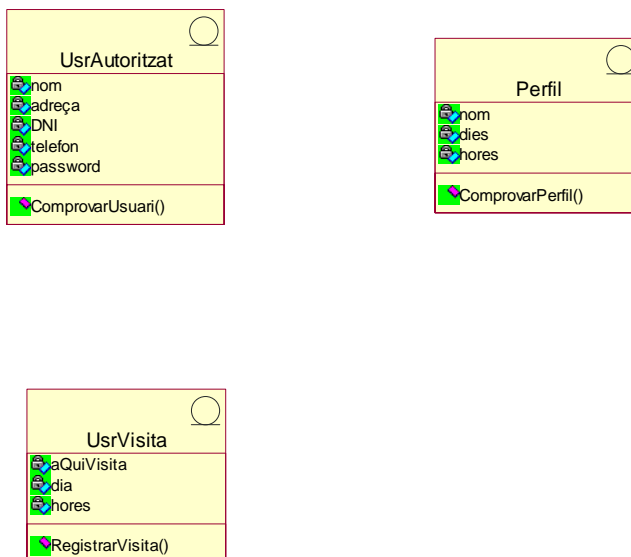
Només es tindran en compte en la correcció les respostes assenyalades en aquesta taula

1	a	b	c	d	e	11	a	b	c	d	e
2	a	b	c	d	e	12	a	b	c	d	e
3	a	b	c	d	e	13	a	b	c	d	e
4	a	b	c	d	e	14	a	b	c	d	e
5	a	b	c	d	e	15	a	b	c	d	e
6	a	b	c	d	e	16	a	b	c	d	e
7	a	b	c	d	e	17	a	b	c	d	e
8	a	b	c	d	e	18	a	b	c	d	e
9	a	b	c	d	e	19	a	b	c	d	e
10	a	b	c	d	e	20	a	b	c	d	e

PREGUNTES CURTES (3 punts)

21) Un sistema de control d'accés s'ha d'encarregar de: 1) Obrir les portes d'un edifici als usuaris que hi volen accedir segons els privilegis assignats i 2) permetre a un usuari administrador, prèvia validació d'un password, gestionar els usuaris. Feu el diagrama de casos d'ús (només del que es descriu).

22) Imaginem les classes del següent diagrama en un disseny d'una aplicació de control d'accés. Tenim els usuaris que poden accedir a un edifici (*UsrAutoritzat*); les visites (*UsrVisita*), que poden considerar-se usuaris eventuais; i els perfils d'usuari (*Perfil*) que indiquen els privilegis corresponents. Dibuixeu les relacions entre les classes que convinguin.



23) Fer el diagrama de seqüència (**MOLT IMPORTANT utilitzar la notació de classes d'anàlisi**) de l'escenari *preparar comanda*. Consisteix a mostrar una llista de comandes pendents. L'encarregat seleccionarà una comanda perquè s'imprimeixi.