

NOME _____ COGNOME _____

Per ogni attività indicare la fase del processo software in cui l'attività avviene:

attività	fase del processo software
definizione dei requisiti funzionali	specificazione
controllo del soddisfacimento dei requisiti prima della consegna	collaudo
definizione dell'architettura del sistema	progettazione
correzione di difetti emersi dopo la consegna del sistema	manutenzione
integrazione dei componenti implementati	collaudo
ricerca di difetti nel sistema prima della consegna	collaudo
scrittura del codice	implementazione
definizione dei requisiti non funzionali	specificazione
aggiunta di nuovi requisiti dopo la consegna del sistema	manutenzione
definizione del controllo e del comportamento dei componenti	progettazione

Considerando un sistema software per gestire gli appelli, indicare tre esempi di requisito funzionale e tre esempi di requisito non funzionale (di cui uno di prodotto, uno organizzativo e uno esterno):

RF: creazione nuovo appello, modifica appello, creazione utente

RNF prodotto: training nell'utilizzo del software

RNF organizzativo: utilizzo di una particolare libreria

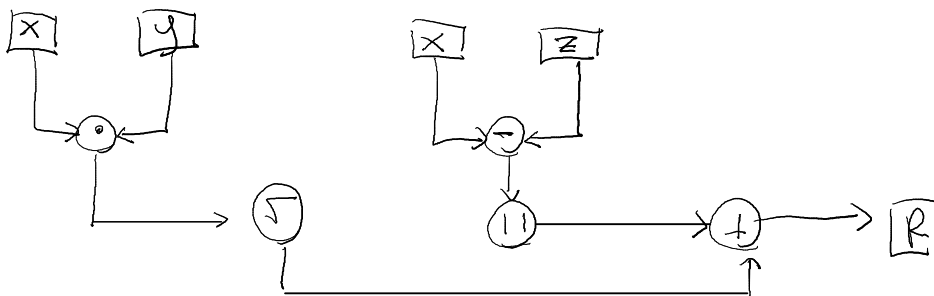
RNF esterno: compatibilità con Moodle

Per ogni attività indicare la fase del processo di specifica in cui l'attività avviene:

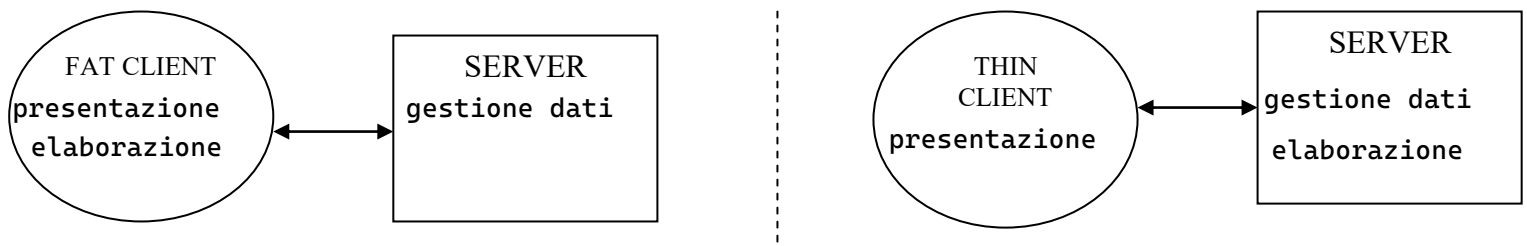
attività	fase del processo di specifica
classificazione dei requisiti	analisi dei requisiti
modellazione dei requisiti	deduzione dei requisiti
controllo della precisione del documento dei requisiti	validazione dei requisiti
valutazione della possibilità di realizzare il sistema	studio di fattibilità
studio del dominio applicativo del sistema	deduzione dei requisiti
dialogo con gli stakeholder	deduzione dei requisiti
valutazione della effettiva utilità del sistema	studio di fattibilità
controllo della completezza del documento dei requisiti	validazione dei requisiti
assegnazione di priorità ai requisiti	analisi dei requisiti

Rappresentare la seguente formula con un modello data-flow in cui x, y, z sono i dati di input iniziali, i nodi filter rappresentano gli operatori matematici, e f è l'output finale:

$$\sqrt{x \cdot y} + |x - z| = f$$



Si consideri l'architettura stratificata su 3 livelli e distribuita su 2 macchine. Indicare nel caso di thin client e fat client quali strati risiedono sulla macchina "client" e quali sulla macchina "server":



Descrivere lo scopo di ogni strato: _____

presentazione: mostrare interfaccia all'utente e ricevere dati in input

elaborazione: elaborare i dati ricevuti dall'utente per produrre dati in output

gestione dati: aggiunta, modifica, rimozione dati da database

Indicare per ogni aspetto se può essere collaudato tramite ispezione e/o testing. Motivare la risposta:

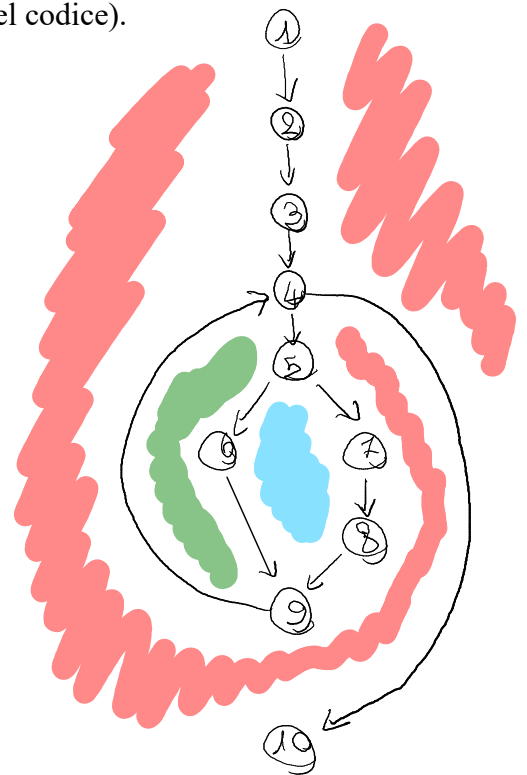
aspetto	testing	ispezione
posizione dei banchi nel codice un test che fallisce può rivelare la posizione dell'errore, così come un'accurata ispezione del codice	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
strutturazione del codice evidente solo durante l'ispezione	<input type="checkbox"/>	<input checked="" type="checkbox"/>
tempo di risposta del sistema misurabile solo tramite test	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Disegnare il flow-graph corrispondente alla funzione (a fianco del codice).

```

1 float calcola_potenza(float base, int esp) {
2     int i=0;
3     float potenza=1;
4     while ( i < abs(esp) ) {
5         if (esp < 0)
6             potenza = potenza / base;
7         else
8             potenza = potenza * base;
9         i++;
10    }
11    return potenza;
12 }

```



Indicare un metodo per calcolare la complessità ciclomatica ed applicarlo al flow-graph ottenuto.

Contare le regioni del codice (distinte dai colori) = 3

Cosa indica la complessità ciclomatica?

Indica il numero minimo di test case richiesti per eseguire almeno una volta ogni parte del codice.

Individuare i cammini indipendenti all'interno del flow-graph.

1. 1,2,3,4,5,6,9,4 ...

2. 1,2,3,4,5,7,8,4 ...

3. 1,2,3,4,10

Per ogni cammino ottenuto definire un test-case che determini tale cammino.

1. Input:(2,-1) Output:1/2

2. Input:(2,2) Output:4

3. Input:(5,0) Output:1

BONUS! Si può mandare il crash il programma con Input:(0,-1) :)

Considerando ancora il sistema per la gestione degli appelli, fornire un esempio concreto di manutenzione correttiva, adattiva, migliorativa.

Correttiva: correzione di un errore che impedisce la modifica della data di un appello

Adattiva: aggiornamento della versione di database utilizzato

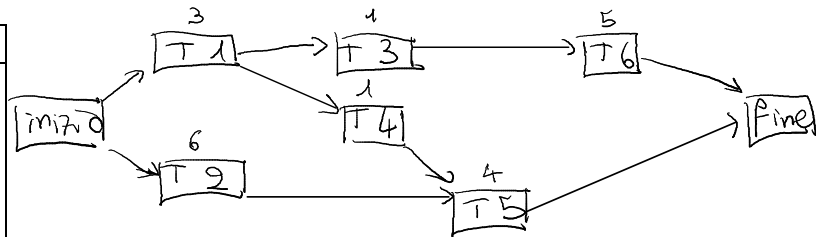
Migliorativa: aggiunta di tema scuro

Indicare per ogni attività se appartiene al processo software oppure alla gestione del progetto:

Attività	gestione del progetto	processo software
Monitoraggio del processo	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Testing	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Stima dei tempi	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Definizione dell'architettura	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Scrittura del codice	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Stima del costo	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Assegnazione delle risorse	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Analisi dei requisiti	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Analisi dei rischi	<input checked="" type="checkbox"/>	<input type="checkbox"/>

All'interno di un processo software sono stati individuati i seguenti task. Disegnare l'activity network (a fianco della tabella).

Task	Durata	Dipendenze
T1	3gg	-
T2	6gg	-
T3	1gg	T1
T4	1gg	T1
T5	4gg	T2, T4
T6	5gg	T3



Individuare il cammino critico e durata complessiva: T2, T5: 10 giorni

Qual è il ritardo massimo consentito a T1 assumendo che gli altri task rispettino i tempi previsti?

Spiegare il motivo: T1 può tardare di un giorno, in quanto i task T1, T3 e T6 impiegano 9 giorni per essere svolti, ed il cammino critico ne dura 10.

Descrivere il possibile effetto di ogni rischio. Quindi classificare il rischio in base a tale effetto:

Rischio	Rischio di prodotto	Rischio di progetto	Rischio di business
Eccessivo ritardo nel reperimento di hardware o software alcune parti del progetto devono essere rimandate	○	⊗	○
Uno sviluppatore lascia l'azienda 1. se era una figura centrale nel progetto, rischio di progetto	⊗	⊗	⊗
2. se è stata licenziato ed ha deciso di vendicarsi, rischio di prodotto E business la prossima volta pensaci due volte prima di fare "restructuring"			
Un prodotto concorrente viene messo sul mercato concorrenza (cos'altro dovrei scrivere?)	○	○	⊗

Associare ogni caratteristica ai modelli di processo in cui è presente:

Caratteristica	sviluppo incrementale	eXtreme Programming	sviluppo evolutivo
la progettazione è fatta solo ad alto livello	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
i test-case sono scelti prima dell'implementazione	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
all'inizio del processo si stabilisce il numero di versioni intermedie del prodotto	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
la versione corrente del sistema si può considerare un prototipo evolutivo	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
si stabilisce all'inizio del processo quali sono i requisiti realizzati da ciascuna versione intermedia	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
la fase di progettazione avviene una sola volta, all'inizio del processo	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
il committente può valutare release intermedie del prodotto	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
ogni versione intermedia del prodotto può realizzare un insieme di nuovi requisiti	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
la specifica è fatta solo ad alto livello	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>

Fornire tre esempi di strumenti CASE (Computer Aided Software Engineering) relativi alle fasi di specifica, progettazione e implementazione, rispettivamente.

Specifica: Visual Paradigm (matrice di tracciabilità)

Progettazione: Visual Paradigm (diagramma di sequenza)

Implementazione: IDE(?)