



SAPIENZA
UNIVERSITÀ DI ROMA

**Esame di Stato – Professione Ingegnere dell’Informazione
IV Prova – 12 luglio 2018
Computer Science - Informatica LM-18**

Prima Traccia

Un condominio vuole adottare un servizio di votazione on-line da applicare per alcune decisioni da prendere sui palazzi che gestisce, come ad esempio la frequenza di potatura delle aiuole e degli alberi, la sostituzione delle porte di ingresso dei vari palazzi, l’installazione di un sistema di sicurezza, ecc.

Al termine di ciascuna riunione di condominio, dopo che l’amministratore ha discusso con i condomini i punti all’ordine del giorno, i vari quesiti associati a ciascun punto vengono caricati dall’amministratore su un sistema informativo, e sono visualizzabili da ciascun condomino attraverso un’interfaccia web accessibile tramite un comune browser. L’accesso ai quesiti richiede l’inserimento di una username e di una password, forniti a ciascun condomino all’atto dell’acquisto dell’abitazione. Ciascun quesito prevede un numero di scelte prefissato. Le votazioni avvengono in una giornata scelta dall’amministratore (dalle 7 di mattina alle 23 del giorno prescelto), ed ogni condomino può votare una ed una sola volta.

Il sistema deve anche prevedere la possibilità che un condomino deleghi il proprio voto ad un altro condomino. Ciò deve essere reso possibile attraverso l’interfaccia web del browser. Nello specifico, quando una votazione viene aperta, un condomino può decidere di delegare il proprio voto accedendo ad una lista contenente tutti i condomini, e scegliendone esattamente uno per la delega. Il condomino delegato riceverà la notifica della delega all’atto della votazione, e potrà decidere se accettarla o rifiutarla. In caso di accettazione, il condomino delegato potrà inserire un voto ai quesiti proposti per ogni delega ricevuta.

Si richiede al candidato di progettare il sistema software descritto, svolgendo nello specifico i seguenti punti e motivando le sue scelte progettuali.

- Si esegua un’analisi dei requisiti, distinguendoli tra funzionali e non funzionali, preferibilmente usando diagrammi di casi d’uso UML.



- Assumendo che il database a disposizione sia relazionale, si costruisca una strutturazione delle informazioni memorizzate utilizzando un diagramma E-R, e si costruisca il corrispondente schema logico in SQL.
- Si progetti l'architettura del sistema, utilizzando preferibilmente diagrammi UML delle classi.
- Si producano in linguaggio SQL le interrogazioni necessarie per soddisfare i casi d'uso.
- Si sviluppi in un linguaggio di programmazione a scelta del candidato (Java, Python, C, ecc.), o eventualmente in pseudocodice, l'algoritmo per l'assegnazione delle deleghe.

Per tutto ciò che non è specificato nel testo, il candidato formuli e giustifichi opportune ipotesi e svolga la prova sulla base di queste.

Seconda Traccia

Un supermercato aperto 24 ore al giorno necessita di coprire con dei turni adeguati tutte le 24 ore di ciascun giorno della settimana. Ci sono 4 turni per giornata, due diurni e due notturni. Ciascun turno dura esattamente 6 ore, e necessita che siano presenti 8 lavoratori a coprire il turno. Il numero totale di lavoratori assunti nel supermercato è di 64 unità.

L'obiettivo del supermercato è lo sviluppo di un sistema informatico che si occupi dell'allocazione dei turni ai lavoratori secondo criteri di equità, considerando che: (1) i turni notturni sono considerati più gravosi di quelli diurni; (2) i turni durante le festività (domenica, Natale, Pasqua, ecc.) sono considerati i più gravosi in assoluto, sempre rispettando il vincolo al punto (1). Inoltre, per un lavoratore, devono essere evitati turni troppo ravvicinati fra loro (con meno di 12 ore fra l'uno e l'altro).

Il sistema informatico si appoggia ad un database centralizzato, che contiene tutti i dati necessari per garantire il servizio di allocazione dei turni. Ad esempio, il database contiene le informazioni anagrafiche dei lavoratori, il loro attuale stato di salute, le informazioni sui turni, e così via.

Il sistema informatico deve anche prevedere un'interfaccia web che consenta ai lavoratori di visionare i turni che gli sono stati assegnati (l'assegnazione avviene con 1 mese di anticipo), prevedendo la possibilità di segnalare in



anticipo un'assenza in caso di ferie (almeno due settimane prima del turno) o di malattia (almeno 24 ore prima del turno previsto). In tal caso, il sistema informatico deve essere in grado di ri-allocare il personale per coprire il turno mancante.

Si richiede al candidato di progettare il sistema software descritto, svolgendo nello specifico i seguenti punti e motivando le sue scelte progettuali.

- Si esegua un'analisi dei requisiti, distinguendoli tra funzionali e non funzionali, preferibilmente usando diagrammi di casi d'uso UML.
- Assumendo che il database a disposizione sia relazionale, si costruisca una strutturazione delle informazioni memorizzate utilizzando un diagramma E-R, e si costruisca il corrispondente schema logico in SQL.
- Si progetti l'architettura del sistema, utilizzando preferibilmente diagrammi UML delle classi.
- Produrre in linguaggio SQL le interrogazioni necessarie per soddisfare i casi d'uso.
- Si sviluppi in un linguaggio di programmazione a scelta del candidato (Java, Python, C, ecc.), o eventualmente in pseudocodice, l'algoritmo che assegna i turni.

Per tutto ciò che non è specificato nel testo, il candidato formuli e giustifichi opportune ipotesi e svolga la prova sulla base di queste.



SAPIENZA
UNIVERSITÀ DI ROMA

**Esame di Stato – Professione Ingegnere dell'Informazione
IV Prova – 12 luglio 2018
Engineering in Computer Science LM-32**

Prima Traccia

Un'agenzia che organizza viaggi vuole memorizzare un insieme di informazioni relative ad alcune mete turistiche che possono essere di interesse per la strutturazione dei viaggi per i propri clienti. In particolare, al fine di migliorare il servizio reso ai propri clienti vuole anche memorizzare i feedback che i clienti riportano sulla qualità dei servizi di cui usufruiscono.

L'agenzia ha interesse a memorizzare i nomi delle città e le relative nazioni di appartenenza. Delle città vuole memorizzare le principali attrazioni che si distinguono in musei, posti di interesse storico-artistico (chiese, piazze, etc.), e posti di interesse naturalistico. Di tutte le attrazioni vuole avere traccia di un nome sintetico, un indirizzo web che fornisca le principali caratteristiche, e di una valutazione della rilevanza dell'attrazione in termini di numero di stelle (da un minimo di 1 ad un massimo di 5). Dei soli musei vuole memorizzare il costo del biglietto di ingresso. Dei posti di interesse naturalistico vuole anche memorizzare il grado di difficoltà necessario per l'esplorazione (da un minimo di uno ad un massimo di 3).

L'agenzia vuole anche suggerire possibili itinerari di visita. Un itinerario è costituito da una sequenza di attrazioni turistiche visitate in un ordine definito. Inoltre, è rilevante memorizzare le strutture ricettive, ed in particolare gli alberghi della città. Oltre al nome, all'indirizzo ed alla categoria dell'albergo, sono di interesse anche i giudizi qualitativi sulla qualità dei servizi forniti che ciascun cliente ha espresso. In particolare i clienti (di cui interessa l'età ed il sesso) possono esprimere un giudizio quantitativo sui servizi dell'albergo in una scala da 1 a 10 (non è di interesse distinguere giudizi espressi dal medesimo cliente in tempi diversi).

Al fine di orientare le proposte di viaggi futuri, l'agenzia vuole memorizzare anche quali attrazioni sono visitate dai propri clienti (di cui si conoscono nome e cognome), con la data di visita. Solo di alcuni clienti che accettano condizioni legate al trattamento privato dei propri dati, l'agenzia dispone



anche di indirizzo, username e password. Di ciascun cliente si conoscono anche i viaggi che ha effettuato, con data di inizio e fine viaggio, attrazioni visitate, strutture ricettive in cui ha soggiornato, e prezzo totale del viaggio. Delle nazioni interessa anche il continente di appartenenza.

Al candidato si richiede di progettare un sistema informatico consistente in un'applicazione mobile che permette ad un cliente di navigare il contenuto informativo sopracitato e di svolgere le seguenti funzioni:

1. Registrarsi al sistema.
2. Effettuare login e logout.
3. Individuare gli le attrazioni maggiormente visitate dai clienti dell'agenzia.
4. Prenotare un viaggio corrispondente ad uno o più itinerari proposti dall'agenzia.
5. Individuare gli itinerari che comprendono la visita di una specifica attrazione.
6. Cancellare una prenotazione.
7. Visualizzare le attrazioni presenti in una determinata città, con tutte le informazioni disponibili.
8. Visualizzare le strutture ricettive in una determinata città, con tutte le informazioni disponibili.

Il sistema deve anche consentire al titolare dell'agenzia di svolgere le seguenti funzioni:

1. Inserimento di nuove strutture ricettive e nuove attrazioni sul sistema.
2. Rimozione di strutture ricettive dal sistema aventi una scarsa valutazione.
3. Gestione efficiente delle prenotazioni: tramite questa funzionalità, si vuole realizzare un sistema che approvi o rifiuti ogni prenotazione effettuata entro 24 ore dalla prenotazione stessa.
4. Gestione del conto e pagamento tramite Paypal.

Sulla base delle specifiche di cui sopra si richiede di:

1. Produrre il diagramma UML dei casi d'uso dell'applicazione
2. Disegnare l'architettura del sistema e discutere possibili strategie di deployment
3. Scegliere tre funzionalità tra quelle descritte e dettagliarne la realizzazione.
4. Realizzare lo schema concettuale della base di dati necessaria allo sviluppo dell'applicazione.
5. Realizzare lo schema logico della base di dati in SQL.
6. Descrivere alcune politiche di ottimizzazione che possono essere utilizzare per l'implementazione della funzionalità "Gestione efficiente delle prenotazioni".

Per tutto quanto non specificato nel testo, il candidato formuli e giustifichi opportune ipotesi e svolga la prova sulla base di queste.



Seconda Traccia

Un dipartimento universitario richiede lo sviluppo di un'applicazione software per la gestione delle missioni dei suoi dipendenti (dottorandi, assegnisti di ricerca, ricercatori, professori, ecc.). Il sistema deve gestire l'intero ciclo di vita di una missione, dalla sua autorizzazione alla liquidazione delle spese di missione.

Tale applicazione verrà utilizzata da:

- i dipendenti che devono effettuare la missione;
- i responsabili dei fondi che devono approvare la missione;
- il personale amministrativo che deve liquidare le spese ritenute ammissibili.

L'applicazione deve consentire l'automazione del seguente processo:

- Un dipendente fa il login sul sistema e richiede l'autorizzazione per una specifica missione, inserendo le date di partenza ed arrivo, la motivazione della missione e i fondi su cui la missione verrà rimborsata.
- L'applicazione identifica il responsabile dei fondi con cui verrà finanziata la missione, e gli invia una richiesta di approvazione.
- Il responsabile dei fondi (anch'esso un dipendente del Dipartimento, potenzialmente lo stesso utente che deve effettuare la missione) può decidere se concedere o meno l'approvazione della missione. In caso di rifiuto, una motivazione deve essere fornita al dipendente.
- Una volta terminata la missione, il dipendente deve caricare sull'applicazione tutte le ricevute e le fatture attestanti il pagamento di spostamenti, pasti e alloggi avvenuti durante la missione.
- Queste informazioni vengono valutate dal personale amministrativo, che verifica l'ammissibilità delle spese sostenute (ad esempio, il rispetto dei massimali per le spese di pasto o di alloggio) e provvede ad effettuare la liquidazione delle spese al dipendente, che viene notificato a pagamento avvenuto.

Il Dipartimento richiede che in una prima versione dell'applicazione siano realizzare le seguenti funzionalità:

1. Gestione dell'intero ciclo di vita della missione, dall'autorizzazione alla liquidazione delle spese.
2. Acquisizione della nota spese e trasmissione dei dati agli uffici amministrativi che gestiscono la liquidazione della missione.
3. Trasmissione dei dati della missione agli uffici che calcolano la retribuzione del dipendente.



4. Realizzazione di un archivio delle missioni che sia in grado di supportare analisi dei dati (es. durata media delle missioni, costo medio, ecc).

Sulla base delle specifiche di cui sopra si richiede di:

1. Produrre il diagramma UML dei casi d'uso dell'applicazione.
2. Disegnare l'architettura funzionale del sistema, mostrando i vari componenti e la loro funzione.
3. Produrre una serie di diagrammi UML (ad esempio, sequence, activity, state) che documentino le funzionalità del sistema.
4. Discutere almeno una tecnica di interazione dei componenti (ad esempio, sincrona VS asincrona) e almeno una tecnologia che potrebbe essere impiegata per la sua realizzazione.
5. Realizzare lo schema concettuale della base di dati necessaria alla realizzazione dell'archivio.
6. Si considerino almeno due requisiti non funzionali e si discuta una possibile strategia per il soddisfacimento di tali requisiti.

Per tutto quanto non specificato nel testo, il candidato formuli e giustifichi opportune ipotesi e svolga la prova sulla base di queste.

ESAME DI STATO PER L'ABILITAZIONE ALLA PROFESSIONE DI INGEGNERE

I SESSIONE 2019

SETTORE DELL'INFORMAZIONE

PROVA PRATICA PER INGEGNERIA GESTIONALE

Il candidato svolga uno dei seguenti temi a scelta:

Tema 1

Si devono organizzare i soccorsi a tre paesi terremotati. I mezzi di soccorso sono raccolti in 4 centri diversi. Nella tabella di seguito sono riassunti la distanza (Km) tra i centri di raccolta ed i paesi, nonché i tempi (in ore) necessari a percorrere un chilometro da parte dei mezzi provenienti da ciascun centro:

	paese 1	paese 2	paese 3	tempi
centro 1	14	19	54	0.1
centro 2	12	31	45	0.2
centro 3	16	45	60	0.15
centro 4	21	44	46	0.25

A ciascun paese è necessario un numero minimo di mezzi di soccorso:

	paese 1	paese 2	paese 3
n° mezzi	18	20	21

mentre da ogni centro di raccolta può essere fornito al più il seguente numero di mezzi:

	centro 1	centro 2	centro 3	centro 4
n° mezzi	34	38	35	22

1. Si costruisca un modello di Programmazione Matematica che verifichi le specifiche assegnate minimizzando i tempi di trasporto.
2. Dire a quale particolare classe di problemi di Programmazione Matematica appartiene il modello. Descrivere una metodologia in grado di risolvere efficientemente il precedente problema di ottimo.
3. Modificare il modello in modo da includere anche le seguenti restrizioni:
 - dal centro 3 si può inviare soccorsi solamente a due paesi su tre;
 - se utilizzano i mezzi del centro 1 non si possono usare i mezzi del centro 2 e viceversa.
4. Dire a quale classe di problemi di Programmazione Matematica appartiene questo nuovo modello e descrivere, nuovamente, una metodologia in grado di risolverlo efficientemente.

Tema 2

Un'industria ha 2 stabilimenti in cui produce 4 differenti prodotti P1, P2, P3, P4.

La tabella che segue riporta la richiesta di ore lavorative per ogni unità di prodotto in ciascuno dei 2 stabilimenti. Nella tabella sono anche riportati i profitti per unità di prodotto (in euro) stimati dall'industria per ogni prodotto.



SAPIENZA
UNIVERSITÀ DI ROMA

**Esame di Stato – Professione Ingegnere dell'Informazione
IV Prova – 12 luglio 2018
Ingegneria Informatica LM-32**

Prima Traccia

Un'agenzia che organizza viaggi vuole memorizzare un insieme di informazioni relative ad alcune mete turistiche che possono essere di interesse per la strutturazione dei viaggi per i propri clienti. In particolare, al fine di migliorare il servizio reso ai propri clienti vuole anche memorizzare i feedback che i clienti riportano sulla qualità dei servizi di cui usufruiscono.

L'agenzia ha interesse a memorizzare i nomi delle città e le relative nazioni di appartenenza. Delle città vuole memorizzare le principali attrazioni che si distinguono in musei, posti di interesse storico-artistico (chiese, piazze, etc.), e posti di interesse naturalistico. Di tutte le attrazioni vuole avere traccia di un nome sintetico, un indirizzo web che fornisca le principali caratteristiche, e di una valutazione della rilevanza dell'attrazione in termini di numero di stelle (da un minimo di 1 ad un massimo di 5). Dei soli musei vuole memorizzare il costo del biglietto di ingresso. Dei posti di interesse naturalistico vuole anche memorizzare il grado di difficoltà necessario per l'esplorazione (da un minimo di uno ad un massimo di 3).

L'agenzia vuole anche suggerire possibili itinerari di visita. Un itinerario è costituito da una sequenza di attrazioni turistiche visitate in un ordine definito. Inoltre, è rilevante memorizzare le strutture ricettive, ed in particolare gli alberghi della città. Oltre al nome, all'indirizzo ed alla categoria dell'albergo, sono di interesse anche i giudizi qualitativi sulla qualità dei servizi forniti che ciascun cliente ha espresso. In particolare i clienti (di cui interessa l'età ed il sesso) possono esprimere un giudizio quantitativo sui servizi dell'albergo in una scala da 1 a 10 (non è di interesse distinguere giudizi espressi dal medesimo cliente in tempi diversi).

Al fine di orientare le proposte di viaggi futuri, l'agenzia vuole memorizzare anche quali attrazioni sono visitate dai propri clienti (di cui si conoscono nome e cognome), con la data di visita. Solo di alcuni clienti che accettano condizioni legate al trattamento privato dei propri dati, l'agenzia dispone



anche di indirizzo, username e password. Di ciascun cliente si conoscono anche i viaggi che ha effettuato, con data di inizio e fine viaggio, attrazioni visitate, strutture ricettive in cui ha soggiornato, e prezzo totale del viaggio. Delle nazioni interessa anche il continente di appartenenza.

Al candidato si richiede di progettare un sistema informatico consistente in un'applicazione mobile che permette ad un cliente di navigare il contenuto informativo sopracitato e di svolgere le seguenti funzioni:

1. Registrarsi al sistema.
2. Effettuare login e logout.
3. Individuare gli le attrazioni maggiormente visitate dai clienti dell'agenzia.
4. Prenotare un viaggio corrispondente ad uno o più itinerari proposti dall'agenzia.
5. Individuare gli itinerari che comprendono la visita di una specifica attrazione.
6. Cancellare una prenotazione.
7. Visualizzare le attrazioni presenti in una determinata città, con tutte le informazioni disponibili.
8. Visualizzare le strutture ricettive in una determinata città, con tutte le informazioni disponibili.

Il sistema deve anche consentire al titolare dell'agenzia di svolgere le seguenti funzioni:

1. Inserimento di nuove strutture ricettive e nuove attrazioni sul sistema.
2. Rimozione di strutture ricettive dal sistema aventi una scarsa valutazione.
3. Gestione efficiente delle prenotazioni: tramite questa funzionalità, si vuole realizzare un sistema che approvi o rifiuti ogni prenotazione effettuata entro 24 ore dalla prenotazione stessa.
4. Gestione del conto e pagamento tramite Paypal.

Sulla base delle specifiche di cui sopra si richiede di:

1. Produrre il diagramma UML dei casi d'uso dell'applicazione.
2. Disegnare l'architettura del sistema e discutere possibili strategie di deployment.
3. Scegliere tre funzionalità tra quelle descritte e dettagliarne la realizzazione.
4. Realizzare lo schema concettuale della base di dati necessaria allo sviluppo dell'applicazione.
5. Realizzare lo schema logico della base di dati in SQL.
6. Descrivere alcune politiche di ottimizzazione che possono essere utilizzare per l'implementazione della funzionalità "Gestione efficiente delle prenotazioni".

Per tutto quanto non specificato nel testo, il candidato formuli e giustifichi opportune ipotesi e svolga la prova sulla base di queste.



Seconda Traccia

Un dipartimento universitario richiede lo sviluppo di un'applicazione software per la gestione delle missioni dei suoi dipendenti (dottorandi, assegnisti di ricerca, ricercatori, professori, ecc.). Il sistema deve gestire l'intero ciclo di vita di una missione, dalla sua autorizzazione alla liquidazione delle spese di missione.

Tale applicazione verrà utilizzata da:

- i dipendenti che devono effettuare la missione;
- i responsabili dei fondi che devono approvare la missione;
- il personale amministrativo che deve liquidare le spese ritenute ammissibili.

L'applicazione deve consentire l'automazione del seguente processo:

- Un dipendente fa il login sul sistema e richiede l'autorizzazione per una specifica missione, inserendo le date di partenza ed arrivo, la motivazione della missione e i fondi su cui la missione verrà rimborsata.
- L'applicazione identifica il responsabile dei fondi con cui verrà finanziata la missione, e gli invia una richiesta di approvazione.
- Il responsabile dei fondi (anch'esso un dipendente del Dipartimento, potenzialmente lo stesso utente che deve effettuare la missione) può decidere se concedere o meno l'approvazione della missione. In caso di rifiuto, una motivazione deve essere fornita al dipendente.
- Una volta terminata la missione, il dipendente deve caricare sull'applicazione tutte le ricevute e le fatture attestanti il pagamento di spostamenti, pasti e alloggi avvenuti durante la missione.
- Queste informazioni vengono valutate dal personale amministrativo, che verifica l'ammissibilità delle spese sostenute (ad esempio, il rispetto dei massimali per le spese di pasto o di alloggio) e provvede ad effettuare la liquidazione delle spese al dipendente, che viene notificato a pagamento avvenuto.

Il Dipartimento richiede che in una prima versione dell'applicazione siano realizzate le seguenti funzionalità:

1. Gestione dell'intero ciclo di vita della missione, dall'autorizzazione alla liquidazione delle spese.
2. Acquisizione della nota spese e trasmissione dei dati agli uffici amministrativi che gestiscono la liquidazione della missione.
3. Trasmissione dei dati della missione agli uffici che calcolano la retribuzione del dipendente.



4. Realizzazione di un archivio delle missioni che sia in grado di supportare analisi dei dati (es. durata media delle missioni, costo medio, ecc).

Sulla base delle specifiche di cui sopra si richiede di:

1. Produrre il diagramma UML dei casi d'uso dell'applicazione.
2. Disegnare l'architettura funzionale del sistema, mostrando i vari componenti e la loro funzione.
3. Produrre una serie di diagrammi UML (ad esempio, sequence, activity, state) che documentino le funzionalità del sistema.
4. Discutere almeno una tecnica di interazione dei componenti (ad esempio, sincrona VS asincrona) e almeno una tecnologia che potrebbe essere impiegata per la sua realizzazione.
5. Realizzare lo schema concettuale della base di dati necessaria alla realizzazione dell'archivio.
6. Si considerino almeno due requisiti non funzionali e si discuta una possibile strategia per il soddisfacimento di tali requisiti.

Per tutto quanto non specificato nel testo, il candidato formuli e giustifichi opportune ipotesi e svolga la prova sulla base di queste.



SAPIENZA
UNIVERSITÀ DI ROMA

Esame di Stato – Professione Ingegnere dell'Informazione
IV Prova – 12 luglio 2018
Ingegneria della Sicurezza e Protezione Civile - LM26

I traccia

In un capannone adibito a Centro Elaborazione Dati (CED) è prevista l'installazione di un sistema di controllo incendi (rivelazione) realizzato mediante sensori ottici puntuali. L'area complessiva del locale è pari a 800 m² e il raggio di copertura di ogni singolo sensore è pari a 8 m.

I sensori che realizzano il sistema di rivelazione provengono da differenti forniture (I, II e III) con probabilità di guasto dichiarate dal fornitore (fornitura I: $p_I = 4.0 \cdot 10^{-3}$; fornitura II: $p_{II} = 2.8 \cdot 10^{-3}$; fornitura III: $p_{III} = 9.0 \cdot 10^{-3}$). La composizione è nota: fornitura I = 28%, fornitura III = 35%.

Il tempo di ritorno dell'evento iniziatore (incendio) è pari a: $T_R(EI) = 10^4$ [anno]

Rispetto all'attivazione efficace del sistema di controllo dell'incendio è richiesto, per ogni posizione del focolaio interna alle aree d'influenza dei singoli sensori e sul contorno delle aree stesse, il funzionamento di **almeno due** dei rivelatori attivi. Ciò implica che lo schema realizzativo del sistema preveda una ridondanza dei dispositivi di rivelazione (sensori) di ordine 2.

Il rischio incendio può essere analizzato rappresentando l'evoluzione del flusso del pericolo (Event Tree Analysis-ETA) sulla base delle indicazioni di seguito riportate.

Per la probabilità dell'evento iniziatore si assuma la condizione di equiprobabilità rispetto alla posizione del focolaio di incendio nell'area in esame.

Gli eventi esaminati per la rappresentazione dell'ETA sono di seguito indicati in successione logico-temporale:

- Sistema di rivelazione ed allarme incendio (di seguito **Sottosistema 1**);
- Sistema di spegnimento automatico (di seguito **Sottosistema 2**), dipendente dal sistema di rivelazione;
- Autosoccorso (fuga degli addetti) dipendente dai sistemi 1 e 2 (di seguito **Sottosistema 3**).

Per l'attribuzione delle probabilità di funzionamento del sistema di spegnimento automatico, si consideri:

- la probabilità di corretto funzionamento del sistema di spegnimento automatico, condizionata dall'attivazione manuale dello stesso - in caso di mancato funzionamento del sistema di rivelazione -, assegnata: $P(E_2 | \bar{E}_1) = 10^{-2} \cdot P(E_2 | E_1)$;

- la probabilità di malfunzionamento del sistema di spegnimento automatico, condizionata dal corretto funzionamento del sistema di rivelazione dell'incendio, è $P(\bar{E}_2 | E_1) = 1.5 \cdot 10^{-3}$;

Ai fini della valutazione dell'efficacia dell'autosoccorso (capacità degli addetti di raggiungere un "luogo sicuro" rispetto alla pianificazione dell'emergenza predisposta nel centro di elaborazione dati) si consideri che l'evento è condizionato dalla



risposta dei sottosistemi preposti alla rivelazione e allo spegnimento (1 e 2).

In caso di corretto funzionamento di entrambi i sottosistemi 1 e 2, la probabilità dell'evento "efficacia dell'autosoccorso" vale: $P(E_3 | E_1 \cap E_2) = 7,8 \cdot 10^{-1}$.

In caso di mancato funzionamento di uno dei due sottosistemi 1 e 2, ai fini della valutazione della probabilità dell'evento "efficacia dell'autosoccorso", si consideri, in prima approssimazione, l'effetto dell'efficacia del sistema di rivelazione dell'incendio equivalente all'effetto dell'efficacia del sistema di spegnimento automatico:

$$P(E_3 | E_1 \cap \bar{E}_2) = P(E_3 | \bar{E}_1 \cap E_2) = 4,5 \cdot 10^{-1}.$$

Infine, in caso di risposta inefficace dei sottosistemi 1 e 2, la probabilità dell'evento "efficacia dell'autosoccorso" vale:

$$P(E_3 | \bar{E}_1 \cap \bar{E}_2) = 5,0 \cdot 10^{-3}.$$

Assumendo che:

- La presenza di addetti (**Evento 4**) all'interno dell'area di lavoro è determinata dallo svolgimento di due turni lavorativi giornalieri:
 - **turno lavorativo: 6.00÷14.00** – con $N = 6$ addetti;
 - **turno lavorativo: 12.00÷20.00** – con $N = 4$ addetti;dove, N indica il numero di addetti presente all'interno dell'area di lavoro durante il turno.
- La **probabilità dell'evento** $\bar{E}_4$ (presenza di addetti nel capannone) sia coerentemente attribuita in funzione della durata dei turni (si assumano le giornate lavorative annue pari a 250).

- 1) Si calcoli la probabilità di malfunzionamento del sistema di controllo incendio rispetto alla posizione del focolaio considerata nell'analisi del problema.
- 2) Si fornisca una rappresentazione grafica dell'albero degli eventi, tenuto conto dell'Evento Iniziatore e dei Sottosistemi attivabili nella gestione del rischio.
- 3) Si valutino le probabilità di fine ramo che caratterizzano l'evoluzione del flusso del pericolo.

Per l'attribuzione del danno associato agli scenari di fine ramo, si consideri che, in caso di **inefficacia dell'autosoccorso** comunque condizionato, l'esito è sempre la morte di tutti i lavoratori presenti. In tutti gli altri casi non si registrano conseguenze per gli esposti.

- 4) Si calcoli il valore atteso della variabile aleatoria danno $E[D]$
- 5) Si determini la distribuzione retrocumulata $F(D) = P(D \geq D^*)$ della variabile aleatoria danno.
- 6) Si rappresenti un tipico schema a blocchi di un sistema di rilevazione e allarme incendi.

