

Simulazione verifica scritta 3 : Selezione in C

(tempo: 1h 30 min. + 27 minuti per studenti con certificazione)

I tre progetti relativi agli esercizi svolti vanno salvati e in una cartella sul desktop chiamata `Cognome_Nome`, il nome di ciascun file deve essere quello relativo all'esercizio svolto con l'aggiunta del cognome (esempio: `ordinatiLaini`, `noleggioLaini`, `piscinaLaini`)

- Il punteggio massimo viene assegnato se l'esercizio viene svolto correttamente (sarà valutato anche il corretto utilizzo di variabili e costanti)
- Punteggio parziale viene assegnato all'esercizio in presenza di bug. (Non è necessario aggiungere controlli ai dati di input se non specificatamente richiesto)
- Zero punti vengono assegnati se l'esercizio non viene svolto o se non fornisce i risultati corretti.

Il professore si riserva di sottoporre lo studente a prova orale chiedendo chiarimenti sul codice scritto se necessario. In tal caso l'orale andrebbe ad integrare la valutazione scritta.

Capacità di creare e salvare correttamente un progetto con IDE CodeBlocks (1 punto)

Esercizio 1: `minuti_secondi` (4 punti)

Scrivi un programma in linguaggio C.

Il programma deve:

- chiedere all'utente di inserire un numero di **ore**;
- chiedere all'utente se vuole convertire le ore in **minuti (min)** oppure in **secondi (s)**.

In base alla scelta dell'utente, il programma deve:

- calcolare i **minuti**, oppure
- calcolare i **secondi**.

Infine, il programma deve mostrare a schermo il risultato della conversione scelta.

Esercizio 2: `fratelli` (2 punti)

Scrivere un programma in C che, date in input le età di tre fratelli, calcoli e mostri la differenza di età fra il fratello maggiore e il fratello minore.

Esercizio 3: `gioco matematico` (3 punti)

Leggi da input **due numeri**.

Poi:

- fai la **somma mentalmente**;
- inserisci nel computer il **risultato della somma** che hai pensato.

Il programma deve:

- calcolare la **somma corretta** dei due numeri;
- confrontare la somma del computer con quella inserita dall'utente.

Se i due risultati sono uguali, il programma deve stampare:

- **“Bravo, hai indovinato al primo tentativo”**

Se i risultati sono diversi, il programma deve stampare:

- **“Hai sbagliato, prova ancora”**

A questo punto:

- l'utente inserisce **di nuovo** il risultato della somma.

Se il risultato è corretto, il programma stampa:

- **“Bravino, hai indovinato al secondo tentativo.”**

Se anche il secondo risultato è sbagliato, il programma stampa:

- **“Ti serve un ripasso di matematica”.**