
FONDAMENTI DI PROGRAMMAZIONE A

Tempo a disposizione: 1 ora 40 minuti

Nome Cognome Matricola

Esercizio 1 (8pt). Scrivere una funzione `hasZeroSum` che prenda come parametri un array di numeri interi e la sua lunghezza e ritorna `true` se esiste una sottosequenza nell'array la cui somma è pari a zero, altrimenti la funzione ritorna `false`. Per esempio, se l'array passato come parametro è `[2, 3, -2, 1, -2, 5]`, la funzione dovrà ritornare `true`.

Esercizio 2 (9pt).

- Scrivere una funzione `removeDuplicates` che, preso come parametro una stringa C-style `str` ritorna una nuova stringa ottenuta eliminando tutti i caratteri duplicati da `str`.
- Scrivere una funzione `main` che legge da tastiera una stringa di massimo 50 caratteri, chiama la funzione `removeDuplicates` passando come parametro la stringa letta ed infine stampa il risultato ritornato dalla chiamata alla funzione `removeDuplicates`.

Esercizio 3 (13pt). Scrivere una funzione di nome `concat` con tipo di ritorno `void` che, presi come suoi parametri due liste semplicemente concatenate `l1` ed `l2` i cui elementi hanno campo informazione di tipo `int`, concateni la lista `l2` alla lista `l1`. Ad esempio, se `l1 = {1,7}` e `l2 = {5,9,12}`, dopo la chiamata alla funzione `concat` passando come parametri `l1` ed `l2`, la lista `l1` diventa `{1,7,5,9,12}`. Trattare in modo opportuno anche i casi in cui `l1` e/o `l2` siano liste vuote. Si scriva inoltre il tipo struttura che modella una lista semplicemente concatenata.