

## 1. Record di attivazione A

Scrivi una funzione in C++ che dato in ingresso due interi  $a$  e  $b$ , calcola la somma di tutti gli interi tra  $a$  e  $b$ . Al solito, scrivi tre versioni: una non ricorsiva, una ricorsiva senza tail recursion e una ricorsiva con tail recursion (potresti dover modificare la segnatura).

Fanne una con passaggio (per  $a$  e  $b$ ) per valore, una per riferimento e una con puntatori.

Scrivi anche un main di esempio in cui chiami la funzione con due valori 1 e 2 e invocala in modo assegni il risultato ad una variabile globale TRE. Non usare alcuna altra variabile globale.

Disegna il record di attivazione per tutte e tre le versioni fino alla massima estensione del record di attivazione. Nel caso di tail recursion, spiega quali ottimizzazioni hai adottato o potresti adottare.

Chiama il file `cognome_nome_es1.c`

## 2. Buffer overflow

Scrivi un esempio di buffer overflow, nel quale accedi in scrittura al di fuori di un buffer (ad esempio di caratteri) e non te ne accorgi. Cosa dovresti fare per accorgertene?

## 3. Uso di typedef + struct

Definisci una struttura per i numeri complessi e con le operazioni  $*$  e  $+$  dei numeri complessi.

Usa typedef e separe il `.h` e il `.c`

## 4. Dangling pointer

Fai un esempio di una funzione che resituisce un numero complesso (come puntatore oppure no) con coordinate  $x$  e  $y$  passate alla funzione. Fai in modo di avere un dangling pointer e risolvilò usando malloc. Usa anche free.