

# Domande su memory layout, stack, heap, etc.

## Domanda 1

Guarda il seguente codice:

```
#include <stdio.h>
#include <stdlib.h>

int main() {
    // Allocazione di memoria dinamica per un intero
    int *heap_var = (int*)malloc(sizeof(int));

    // Verifica che malloc abbia avuto successo
    if (heap_var == NULL) {
        fprintf(stderr, "Errore di allocazione della memoria.\n");
        return 1; // Esci se l'allocazione fallisce
    }

    // Assegna un valore all'area di memoria allocata
    *heap_var = 42;

    // Stampa il valore
    printf("Il valore allocato nell'heap è: %d\n", *heap_var);

    // Non chiamando free(heap_var), si verifica un memory leak

    return 0; // La memoria allocata non viene liberata
}
```

Nella funzione main, viene utilizzata la funzione malloc per allocare memoria dinamicamente. In quale sezione della memoria viene allocata la variabile puntata da heap\_var? Cosa accade alla memoria allocata dopo l'esecuzione del programma se la funzione free non fosse stata chiamata? Il termine memory leak ti dice qualcosa?

## Domanda 2

Descrivi le principali differenze tra il segmento Heap e il segmento Stack. In particolare, fai riferimento a come avviene l'allocazione e la deallocazione in ciascuno di essi facendo

riferimento a C o a Java, tipologia di dati che vi risiedono, come vengono allocate e deallocate le variabili in ciascuna delle due aree.

## **Domanda 3**

Cos'è e quando si verifica uno stack overflow? Spiega in quali situazioni può verificarsi

## **Domanda 4**

Cosa significa “allocazione dinamica della memoria”? Fornisci un esempio pratico in C e in Java di come utilizzare malloc() e new.

## **Domanda 5**

Cosa significa “out of memory”?