

Struttura di un processo

Leonardo Bizzoni

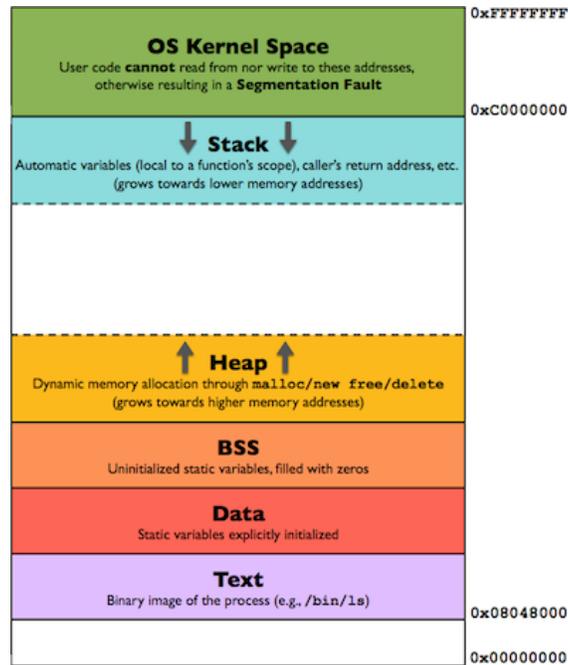
January 26, 2024

Un processo è un **programma** in esecuzione. Lo stato dell'attività di un processo è rappresentato dal PC e dal contenuto dei registri del processore.

La struttura di un processo(**immagine**) in memoria è suddivisa in più sezioni:

- **Text segment**: contiene le istruzioni eseguibili del programma. Solitamente posizionato in memoria read-only all'inizio dello **spazio di indirizzamento** (*address space*)
- **Initialized data segment**: contiene le variabili statiche e globali inizializzate dal programma.
 - **Block Started by Symbol BSS**: situato dopo il Data segment e contiene variabili statiche e globali non inizializzate.
- **Heap**: memoria allocata dinamicamente durante l'esecuzione del programma. Comincia dopo BSS e cresce ad indirizzi di memoria più grandi (*di solito*).
- **Stack**: memoria temporaneamente utilizzata durante le chiamate di funzioni ed assegnamento di variabili locali. Comincia prima del Kernel Space e cresce ad indirizzi di memoria più piccoli fino ad incontrare la fine dell'Heap (*di solito*).

Ovviamente processi distinti hanno immagini distinte.



1 In che direzione cresce lo stack?

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

void growth(int *var) {
    int x;

    if (var != NULL) {
        growth(&x);
    } else {
        if (&x >= var) {
            printf("Cresce verso l'alto (da indirizzi piccoli a grandi).\n");
        } else {
            printf("Cresce verso il basso (da indirizzi grandi a piccoli).\n");
        }
    }
}

int main(void) {
```

```
    growth(NULL);  
}
```