

Vectori STL

(Standard Template Library)

La fel ca un array, vectorii folosesc zone continue de memorie pentru locația elementelor, ceea ce înseamnă ca elementele pot de asemenea fi accesate folosind pointeri către acestea, si la fel de eficient ca într-un array. Ceea ce ii deosebește de un simplu array , este faptul ca dimensiunea lor poate fi schimbata dinamic, aceste schimbări fiind manipulate automat de către container.

Alocare:

Intern, vectorii folosesc un array alocat dinamic ca sa memoreze elemente. Acest array ar trebui sa fie realocat pentru ca sa poată crește in dimensiune, ceea ce implica alocarea unui nou array si mutarea tuturor elementelor in acesta, operație care este extrem de ineficienta, din acest motiv vectorii folosesc extra spațiu ca sa fie pregătiți pentru o creștere a numărului de elemente. Realocările se petrec doar la intervale de creștere logaritmice de memorie, ceea ce duce la o complexitate amortizata constanta de memorie.

Comparativ cu un array simplu, vectorii folosesc mai multa memorie pentru a putea manipula creșterea într-un mod dinamic si mai eficient. Comparativ cu listele, vectorii sunt cu mult mai ineficienți la adăugarea elementelor in alte poziții decât la final.

Dealocare:

Una dintre cele mai simple metode de dealocare a vectorilor este de a îl declara pe stack. Destructorul va fi apelat in momentul execuției metodei, iar destructorul se va asigura ca memoria alocata este eliberata.

```
void Dealocare() {  
    std::vector<int> v;  
    ...  
}
```

Pushback()

Complexitatea tinde către $O(1)$, garantata de standardul `std::vector`, datorita realocărilor care se petrec doar la un interval logaritm de creștere a memoriei, deși `pushback`-ul este mai încet, întrucât accesează de mai multe ori aceleași linii de Cache atunci când are loc realocarea de memorie(mutarea elementelor vechiului array intr-unul nou).

Size():

Returnează numărul de elemente dintr-un vector.

Capacity()

Returnează memoria alocată curent vectorului, exprimată ca număr de elemente raportate la memoria alocată vectorului.

() vs []

() este folosit pentru a transmite operatori unei funcții, iar [] este o prescurtare pentru pointerul către un anumit element din array.

Ex: $a[x] = *(a+x)$