Kontejner dek **<deque>**

Ko ovo ne zna, kečina.

Deklaracija: **deque <tip\_podatka> ime\_deka;**

Dabl ended Red/Kju je kontejner / struktura podataka prvenstveno namenjena smeštanju podataka koji imaju sledeće osobine :

1. Red je nalik nizu, ali u kome je moguće pristupiti samo prvom i poslednjem članu reda.
2. Članovi se dodaju i na početak i na kraj reda
3. Članovi se skidaju i sa početka i sa kraja reda.

Dek je moguće implementirati i uz pomoć sopstvenog niza. Za nijansu je više koda, ali se dobija na brzini (empirijski oko 50%).

**deque** **<int> Q;**

Ako je potrebno saznati koliko članova ima u dek-u sledeća funkcija to daje: **Q**.**size**();

Dodavanje člana u deque (na kraj dek-a): **Q**.**push\_back**(vrednost);

Dodavanje člana u deque (na početak dek-a): **Q**.**push\_front**(vrednost);

Skidanje člana iz deque (sa početka dek-a): **Q**.**pop\_front**();

Skidanje člana iz deque (sa kraja dek-a): **Q**.**pop\_back**();

Pristup prvom članu dek-a: **Q**.**front**();

Pristup poslednjem članu dek-a: **Q**.**back**();

Ako je potrebno pražnjenje dek-a u toku izvršavanja programa to mora peške odpopovati;  
  
To i ostale metode i funkcije vezane za dek kontejner imate na [**tutorijalu za C++ ←(click here)**](http://www.cplusplus.com/reference/deque/deque/)

* Primer upotrebe dek-a pri pronalazenju RMQ-a na konstantnoj širini intervala : <http://articles.leetcode.com/sliding-window-maximum/>