

## Stek (Stack)

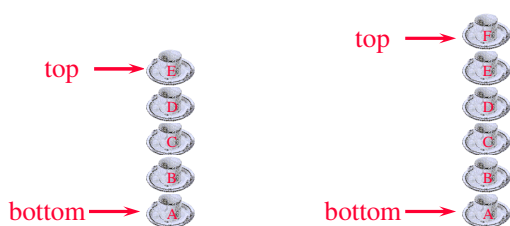


## Stek (Stack)



- Linearna lista.
- Jedan kraj liste je **vrh** ili **top**.
- Drugi kraj je **kraj** ili **bottom**.
- Dodavanje i uklanjanje elemenata obavlja se samo sa strane **top**.

## Stek šolja

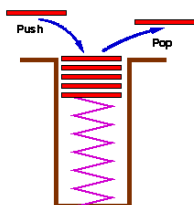


- Dodavanje šolje na stek.
- Uklanjanje šolje sa steka.
- Stek je LIFO (Last In First Out) lista.

## Poklapanje zagrada

- $((a+b)*c+d-e)/(f+g)-(h+j)*(k-l)/(m-n)$ 
  - Izlazni parovi su  $(u,v)$  takvi da se lijeva zagrada na poziciji  $u$  poklapa sa desnom zagradom na poziciji  $v$ .
    - $(2,6)$   $(1,13)$   $(15,19)$   $(21,25)$   $(27,31)$   $(0,32)$   $(34,38)$
- $(a+b)*((c+d)$ 
  - $(0,4)$
  - desna zagrada na poziciji 5 nema odgovarajuće lijeve
  - $(8,12)$
  - lijeva zagrada na 7 nema odgovarajuće desne

### Primjer steka



### Poklapanje zagrada

- Skeniramo ulazni niz s lijeva udesno
- Kada naiđemo na lijevu zagradu, dodamo njenu poziciju na stek
- Kada naiđemo na desnu zagradu, uklanjamo odgovarajuću poziciju sa steka

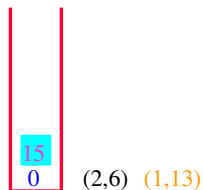
### Primjer

- $((a+b)*c+d-e)/(f+g)-(h+j)*(k-l)/(m-n)$



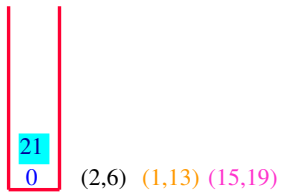
### Primjer

- $((a+b)*c+d-e)/(f+g)-(h+j)*(k-l)/(m-n)$



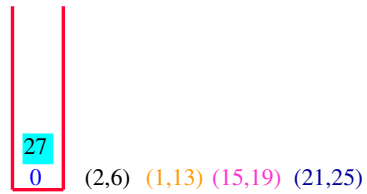
### Primjer

- $((a+b)*c+d-e)/(f+g)-(h+j)*(k-l)/(m-n)$



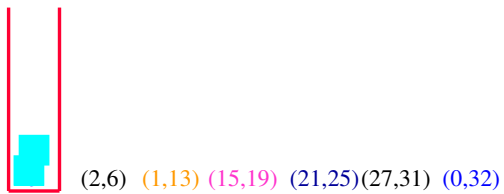
### Primjer

- $((a+b)*c+d-e)/(f+g)-(h+j)*(k-l)/(m-n)$



### Primjer

- $((a+b)*c+d-e)/(f+g)-(h+j)*(k-l)/(m-n)$

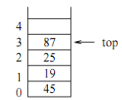


- Itd.

### Implementacija steka

- Pomoću niza

Array/Vector

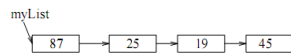


- IsEmpty
- Push(Item)
- Pop

## Implementacija steka

- Pomoću olančane liste

Linked List



Node \* myList;

- StackEmpty
- Push(Item)
- Pop()