



INFINITY.UZ

Biz bilan 0 dan cheksizlikkacha

PYTHON STACK

Presented by: Alimardon Boqijonov



Dars Tartibi

- *Stack nima?*
- *Stack turlari*
- *Pythonda Stack*
- *Stack real hayotda*
- *Algoritmlash*
- *Uy ishi*



Stack nima?

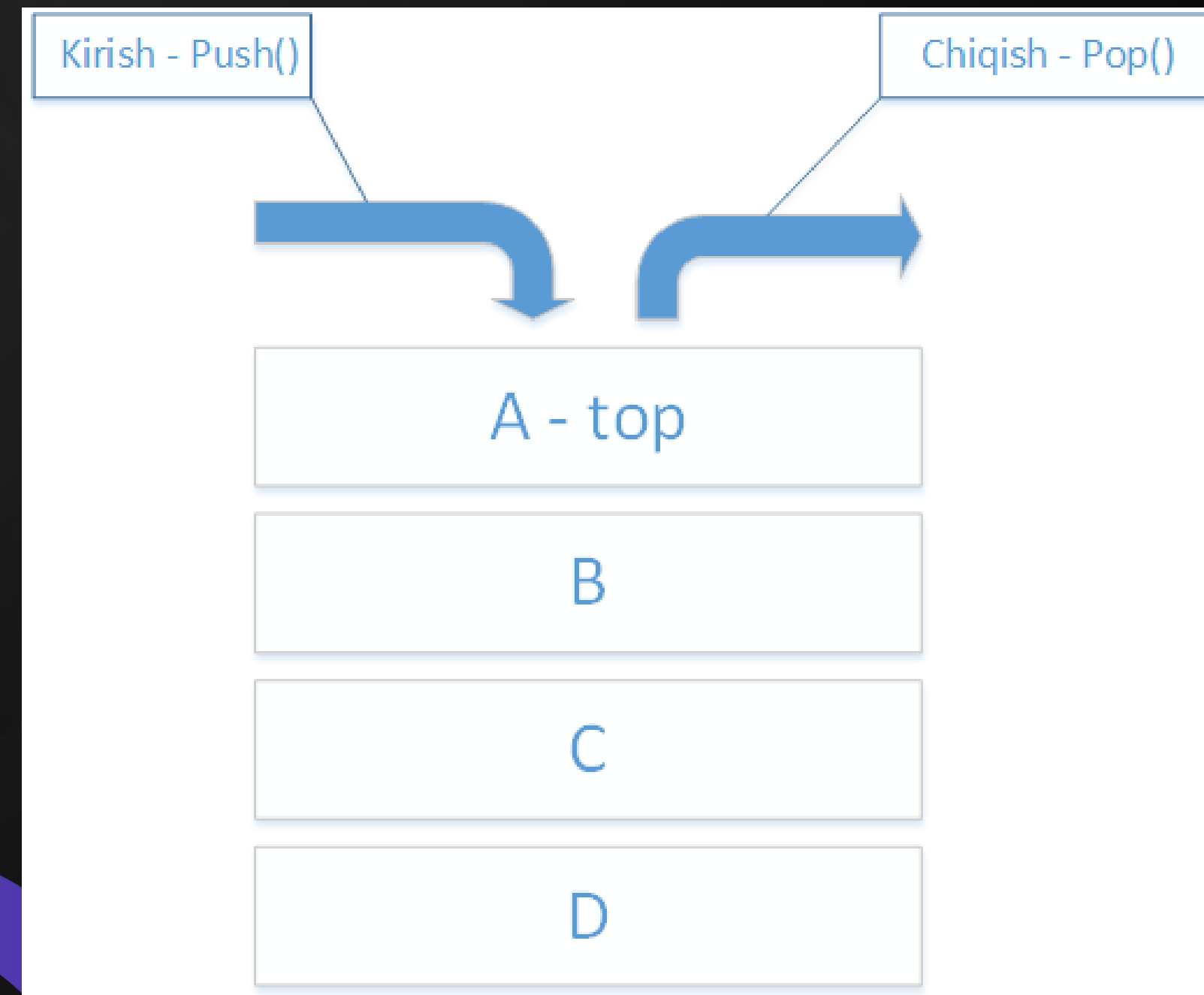
(1)

- Oshxonadagi likopchalar turadigan quti, brovserning orqaga("nazad") tugmasi, ixtiyoriy matn muxarriridagi bekor qilish("CTRL-Z") amali, bularning barchasi Stek ma'lumotlar strukturasi misoldir. "LIFO" y'ani oxirgi kegan birinchi ketadi qoidasi asosiga qurilgan bo'lib kompyuter olamida eng ko'p ishlatiladigan ma'lumotlar strukturasi biri. Demak, bugun **Stek**(Stack) ma'lumotlar strukturasi o'rganamiz.



Stack nima? (2)

- Rasmda ko'rinib turganidek, stek bu obyektlarning ro'yxati bo'lib, eng oxirgi qo'shilgan obyekt xar doim birinchi bo'lib qaytadi. Masalan, ushbu rasmdagi stekka yangi qiymat qo'shilsa A bitta pastga tushadi, yangi qo'shilgan obyekt "top" o'rinni egallaydi. Keyingi murojaat vaqtida aynan yangi qo'shilgan obyekt qaytariladi.
- Ushbu ro'yxatda Stekning tarkibida bo'lgan operatsiyalar berilgan.
- Push(value) stekka yangi qiymat qo'shadi. Aytib o'tilganidek, yangi qaiymat "top" o'ringa borib tushadi.
- Pop()eng oxirgi qo'shilgan obyekt qaytariladi va stekdan o'chiriladi.
- Peek()eng oxirgi qo'shilgan qaytariladi leki stekdan o'chirilmaydi.
- IsEmpty() stek bo'shmi? degan savolga javobgar metod.
- Size() stekdagi obyektlar sonini qaytaradi.



python

```
class Stack:
    def __init__(self):
        self.stack = []

    def is_empty(self):
        return len(self.stack) == 0

    def push(self, item):
        self.stack.append(item)

    def pop(self):
        if not self.is_empty():
            return self.stack.pop()

    def peek(self):
        if not self.is_empty():
            return self.stack[-1]

    def size(self):
        return len(self.stack)
```

- `is_empty()`: Stackning bo'shligini tekshiradi.
- `push(item)`: Stackga yangi element qo'shadi.
- `pop()`: Stackning yuqori elementini o'chirib tashlaydi.
- `peek()`: Stackning yuqori elementini qaytaradi, lekin o'chirmaydi.
- `size()`: Stackdagi elementlar sonini qaytaradi.

Ushbu kod orqali Stack obyektini yaratishingiz va uni qo'llab-quvvatlashingiz mumkin. Misol uchun:

python

 Copy code

```
# Stack obyektini yaratish
stack = Stack()

# Elementlarni qo'shish
stack.push(10)
stack.push(20)
stack.push(30)

# Elementlarni o'qish
print(stack.peek()) # 30
print(stack.pop())  # 30

# Stackning hozirgi o'lchami
print(stack.size()) # 2
```

SAVOL:

**STACK NI QAYSI
HOLATLARDA ISHLATISH
MUMKIN BO'LADI?**



Javob: Stack amaliyotlarida quyidagi holatlarda ishlatish mumkin:

1. Matematikda, oddiy arifmetik amalga oshirishni ifodalash uchun. Misol uchun, infix ifodaning postfix ifodaga aylantirishda, operatorlarni va operandlarni saqlash uchun Stack ishlatiladi.
2. Dasturlashda, funksiyalar bajarilishi davomida foydalanuvchi tomonidan kiritilgan o'zgaruvchilarni saqlash uchun. Funksiyadan chiqqan bo'lganda o'zgaruvchilar ro'yxati qaytariladi.
3. Reversal operatsiyalarida, masalan, matnning teskari qilinishi uchun ishlatiladi.
4. Tarixiy ma'lumotlarni saqlashda. Masalan, brauzerlarda saytlarni ochib bo'lish tartibida foydalaniladi.
5. Tarmoq protokollari yordamida paketlar o'tkazilayotganida, paketlar olish tartibida saqlanadi.

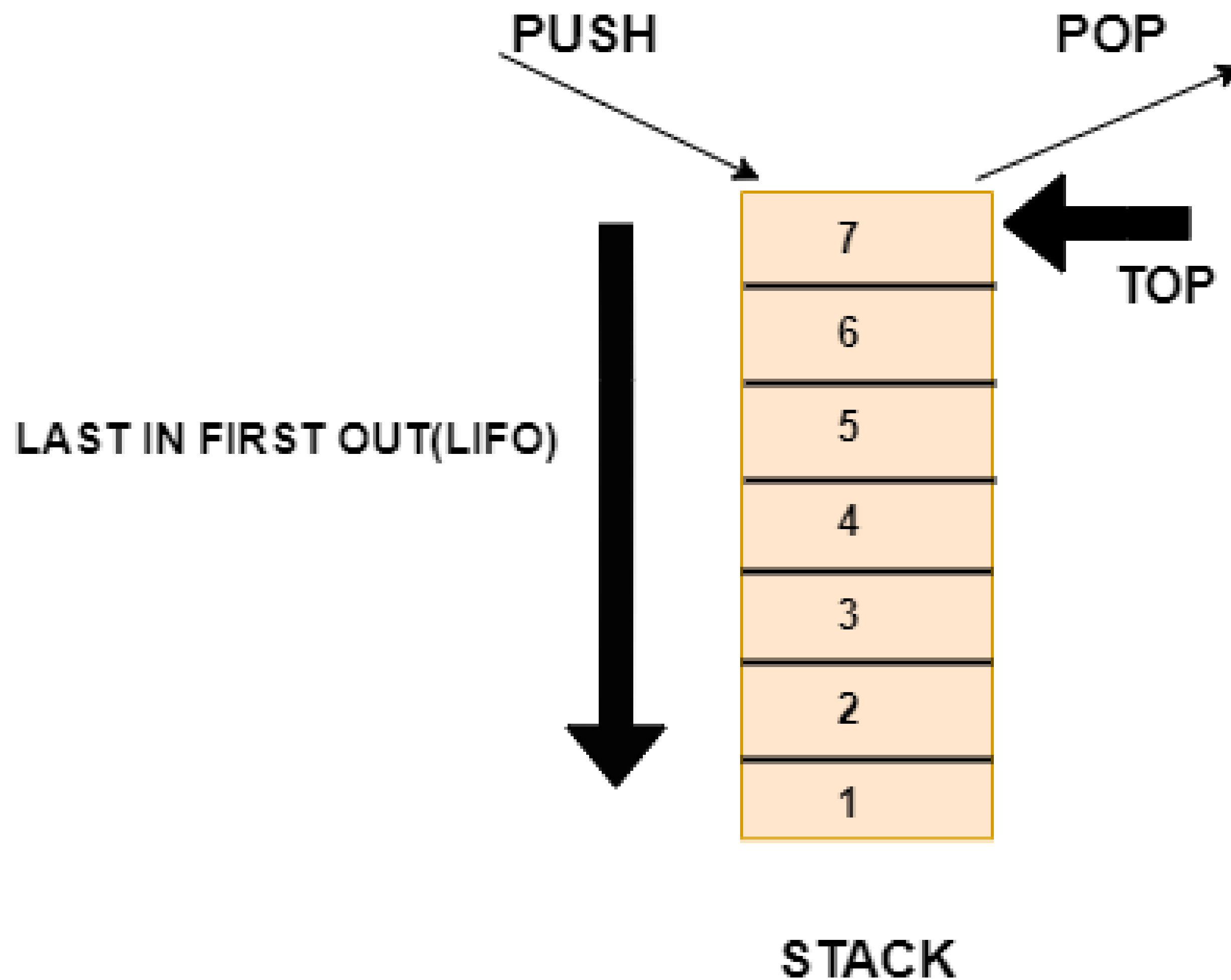
Umid qilamiz, bu javob sizga yordam berdi. Agar yana savollar bo'lsa, bizga yozishingiz mumkin. Muvaffaqiyatli ishlashni tilaymiz!



Savol

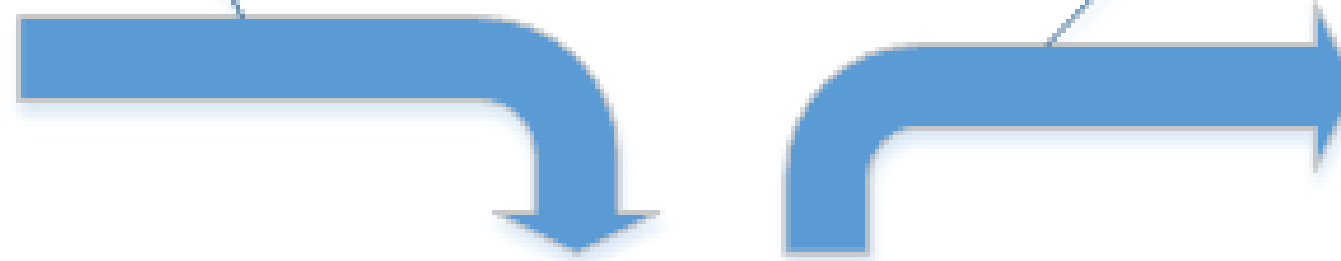
- *Stack obyektini yaratamiz.*
- *Uning push() metodi yordamida odamlarning ismini qo'shamiz.*
- *pop() metodi yordamida eng so'nggi kiritilgan ismni o'chirib olamiz.*





Kirish - Push()

Chiqish - Pop()

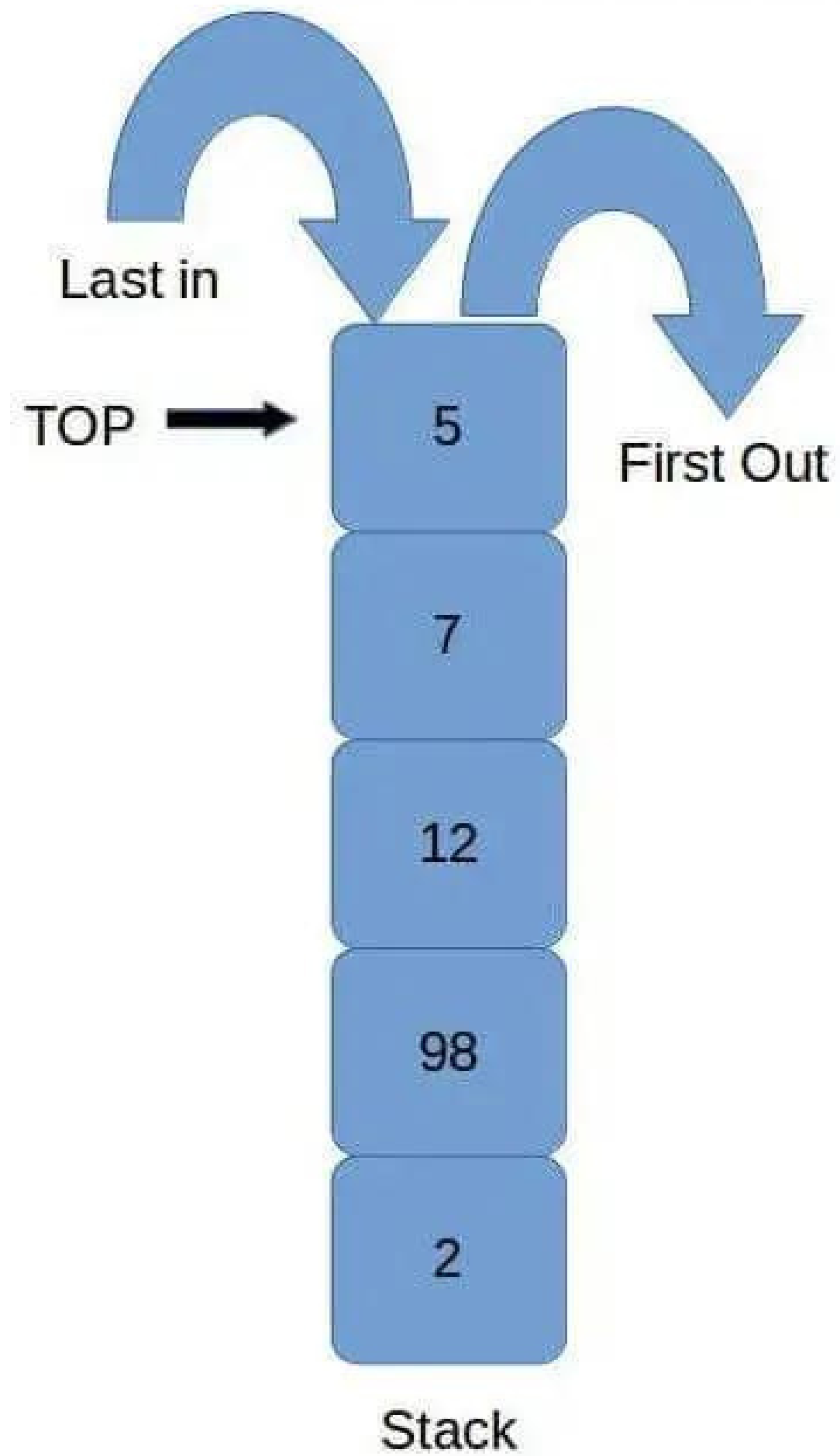


A - top

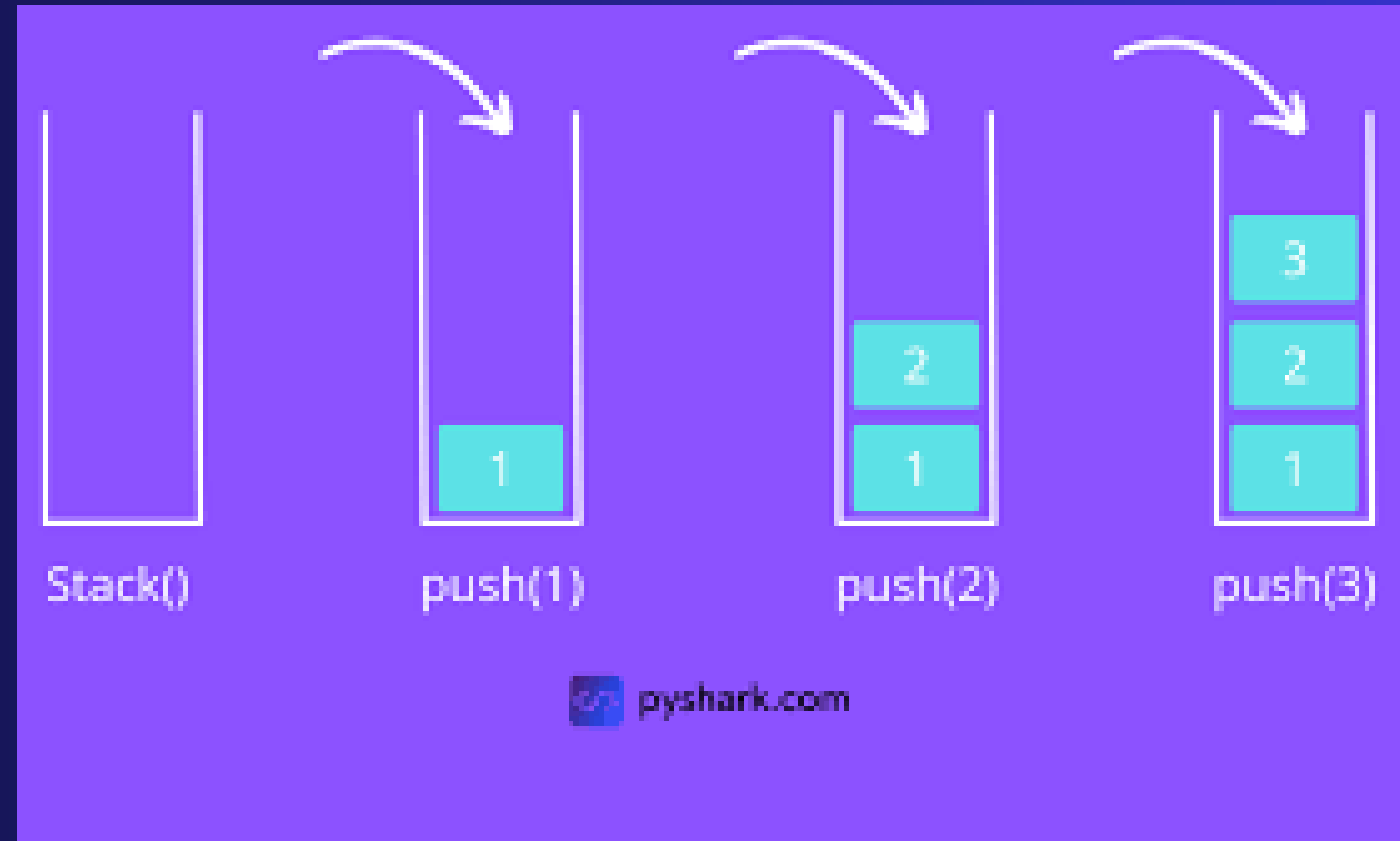
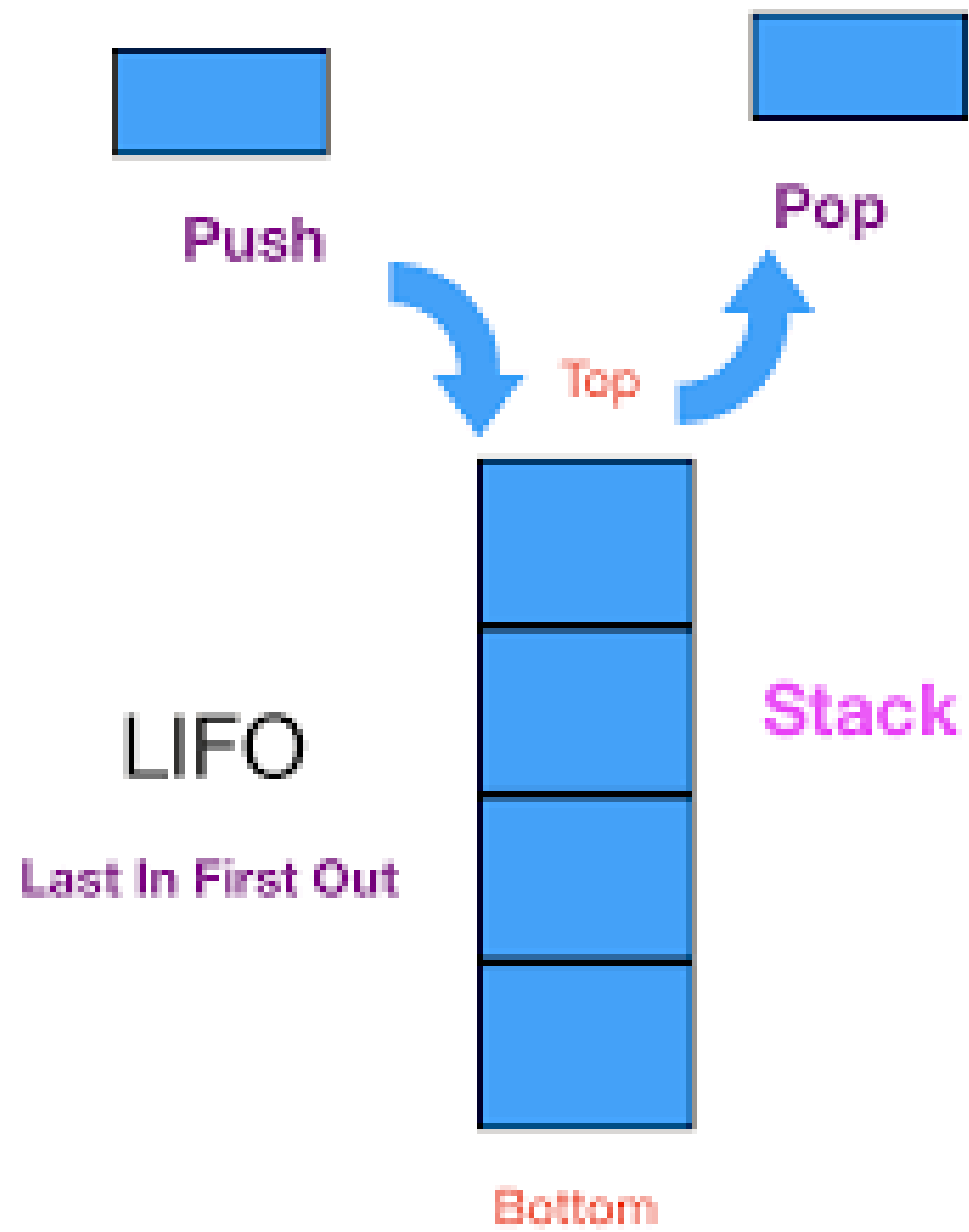
B

C

D



**FIFO-First in
First out**





INFINITY.UZ

BIZ BILAN O DAN CHEKSIZLIKKACHA

THANK YOU

For watching this presentation

