Семинар 11

Введение в программирование на Python

Папулин С.Ю.
papulin_hse@mail.ru

2015

Семинар 10

1. Поиск элемента
2. Последовательный поиск
3. Бинарный поиск

План семинара 11

1. Структуры данных
   - Стек
   - Очередь
   - Дек
2. Принципы объектно-ориентированного программирования (ООП)
3. Класса в Python

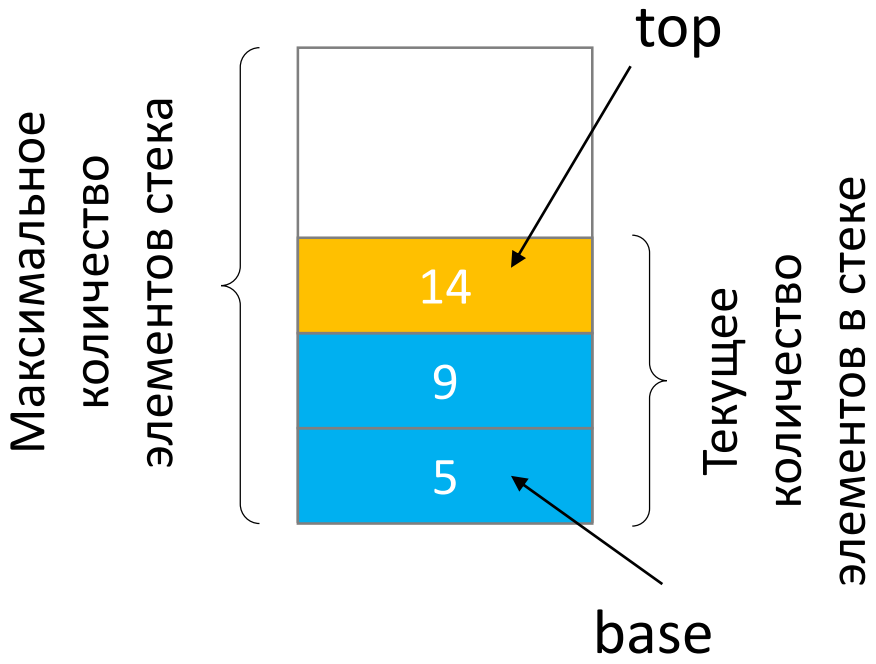
Структуры данных

Стек

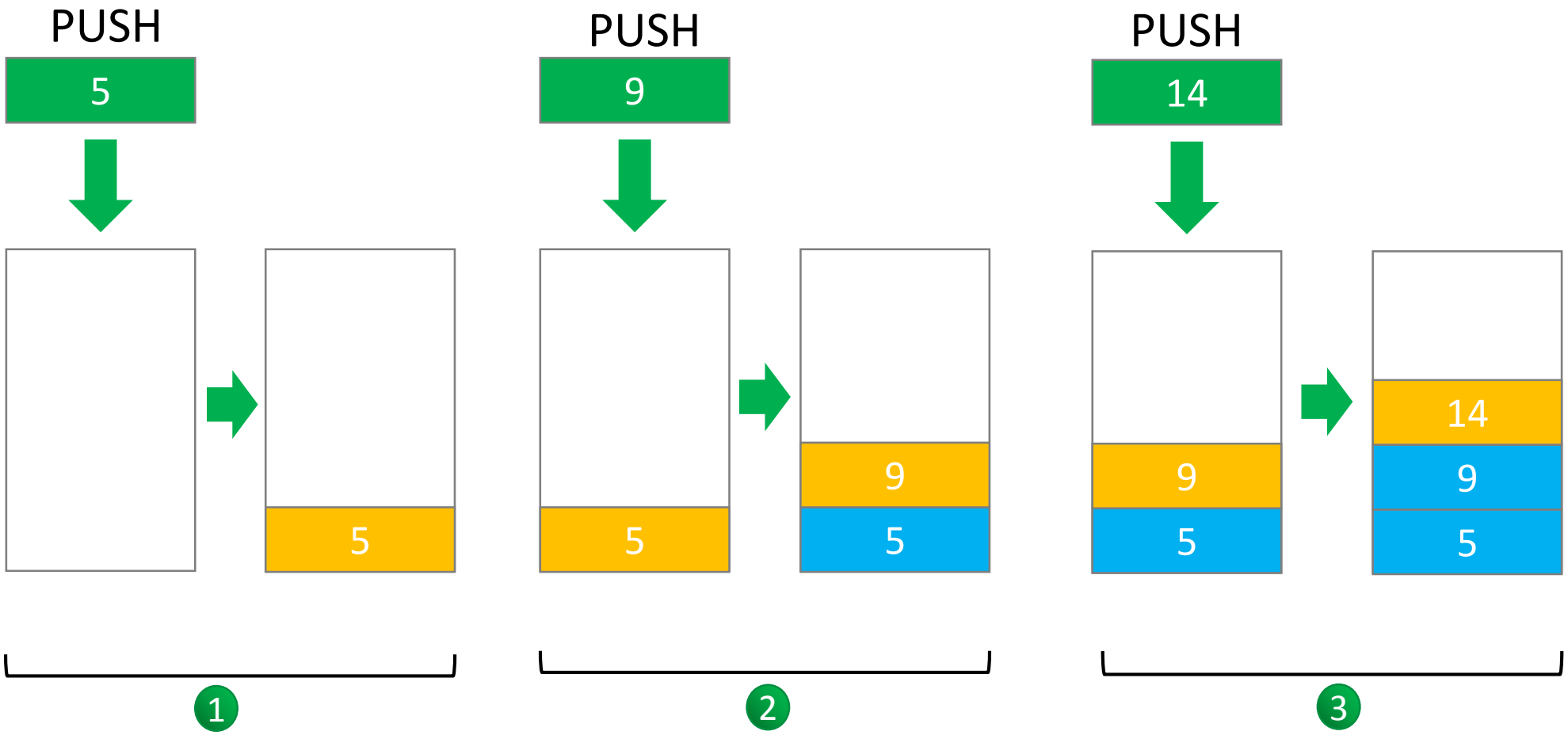
Стек – коллекция элементов, подчиняющаяся правилу «последний пришел – первый вышел» (last-in first-out – LIFO)

Основные операции:

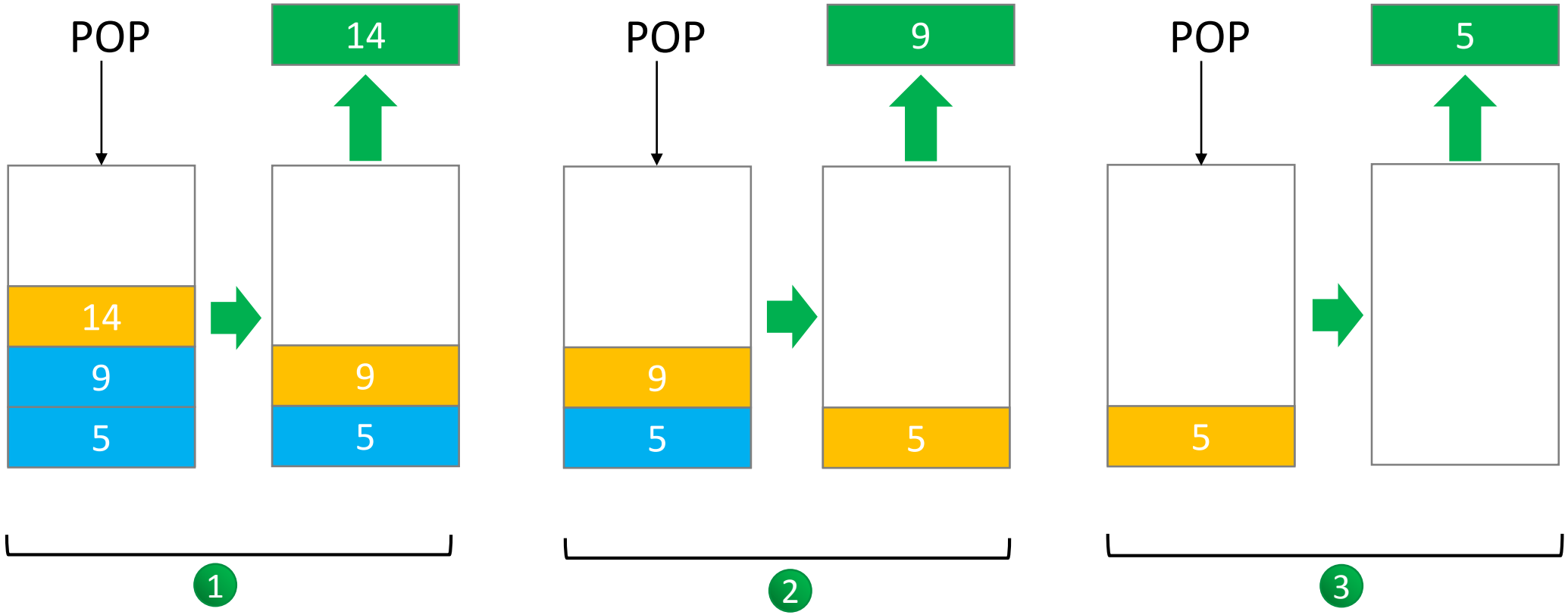
- Добавить/вставить (Push)
- Извлечь (pop)
- Показать последний элемент (back/peek/top)
- Размер стека (size)



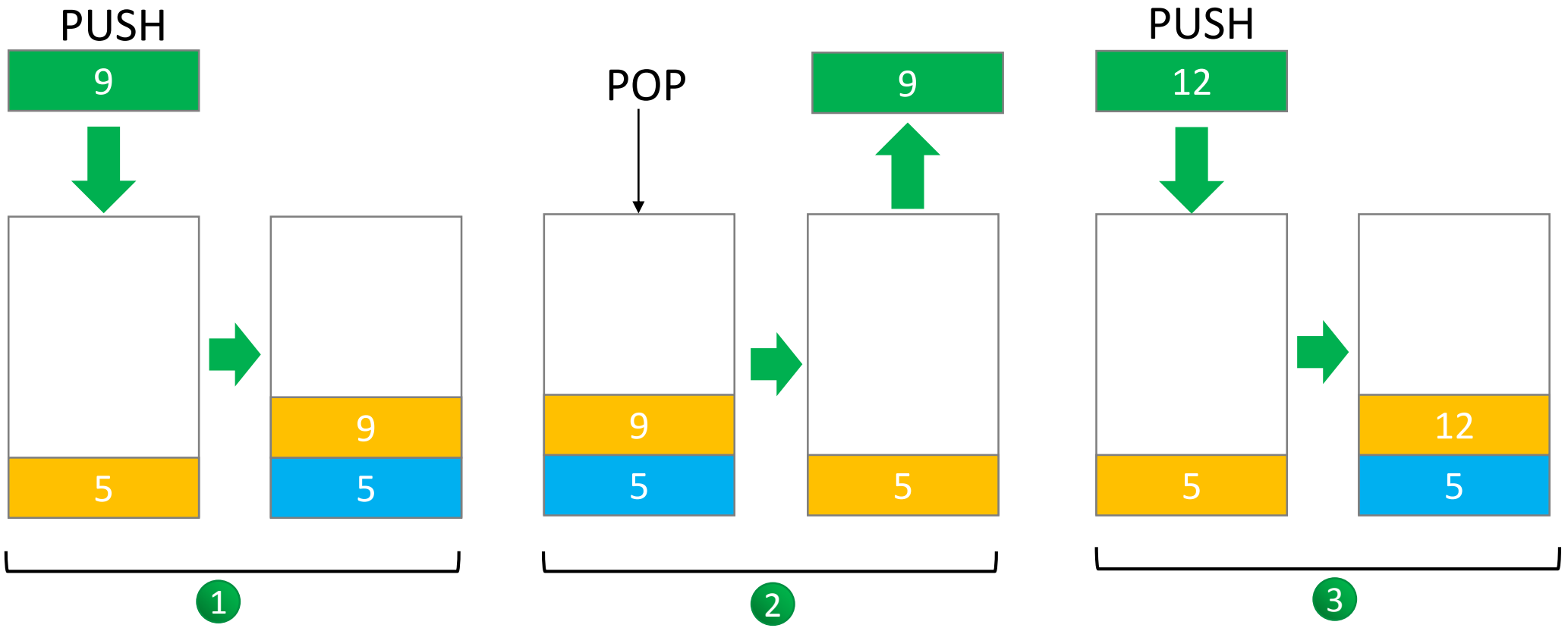
Вставка в стек / PUSH



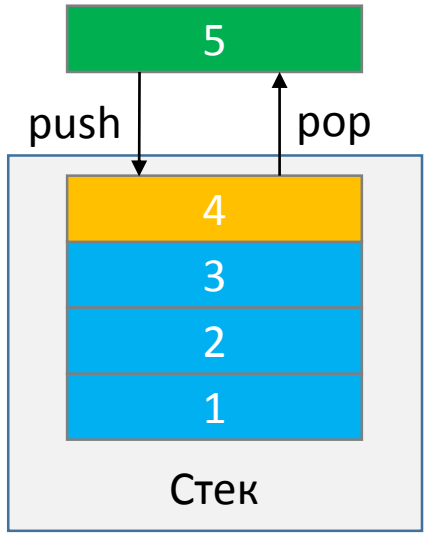
Извлечение из стека / POP

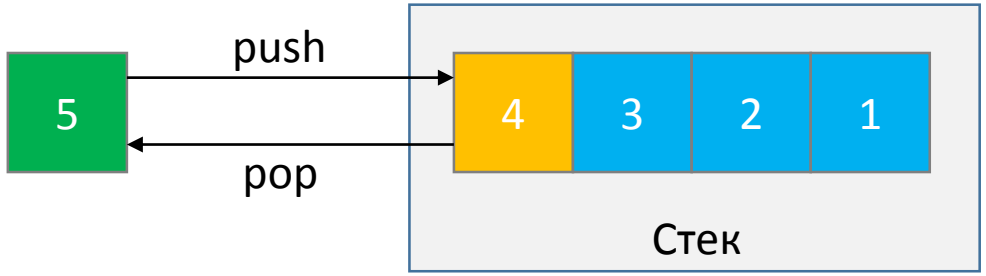


PUSH/POP



Стек



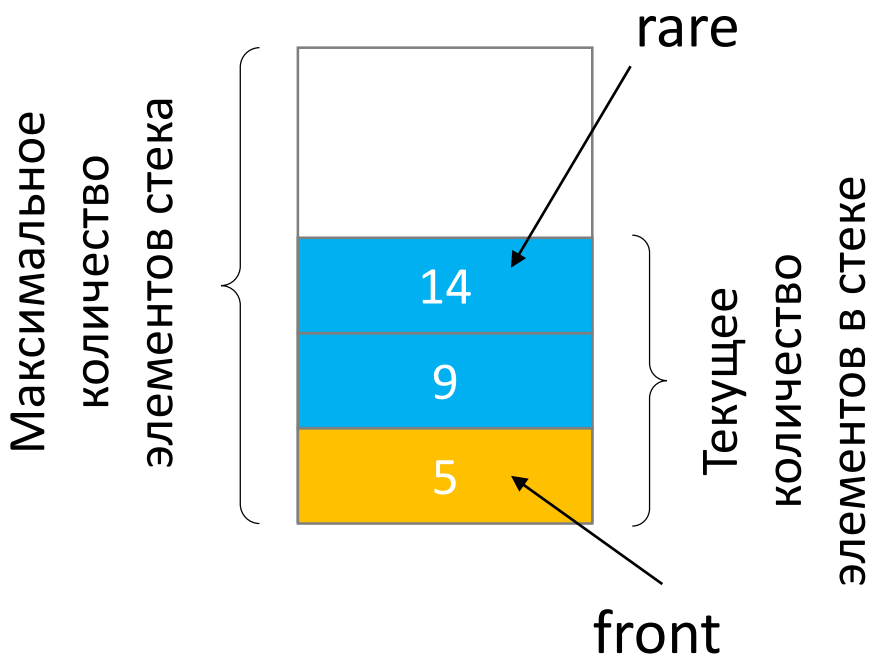


Очередь

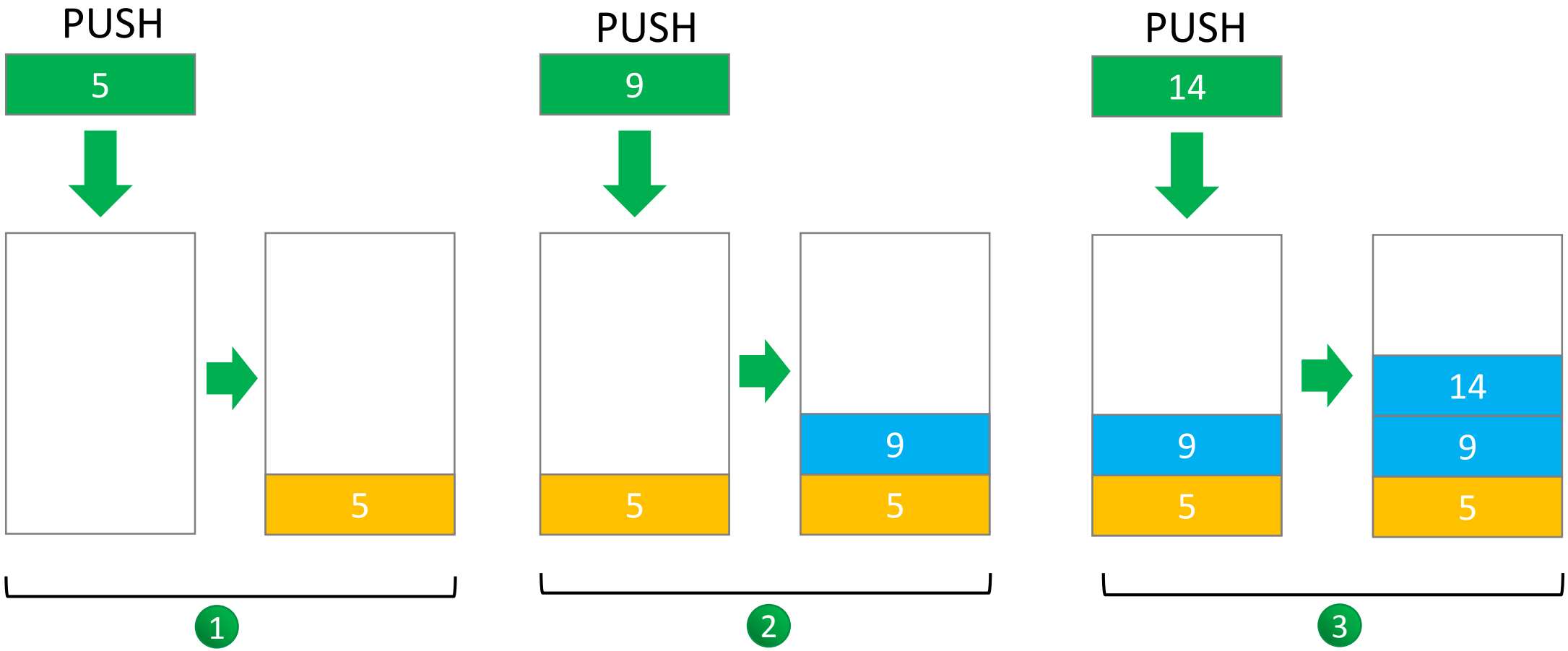
Очередь – коллекция элементов, подчиняющаяся правилу «первый пришел – первый вышел» (first-in first-out – FIFO)

Основные операции:

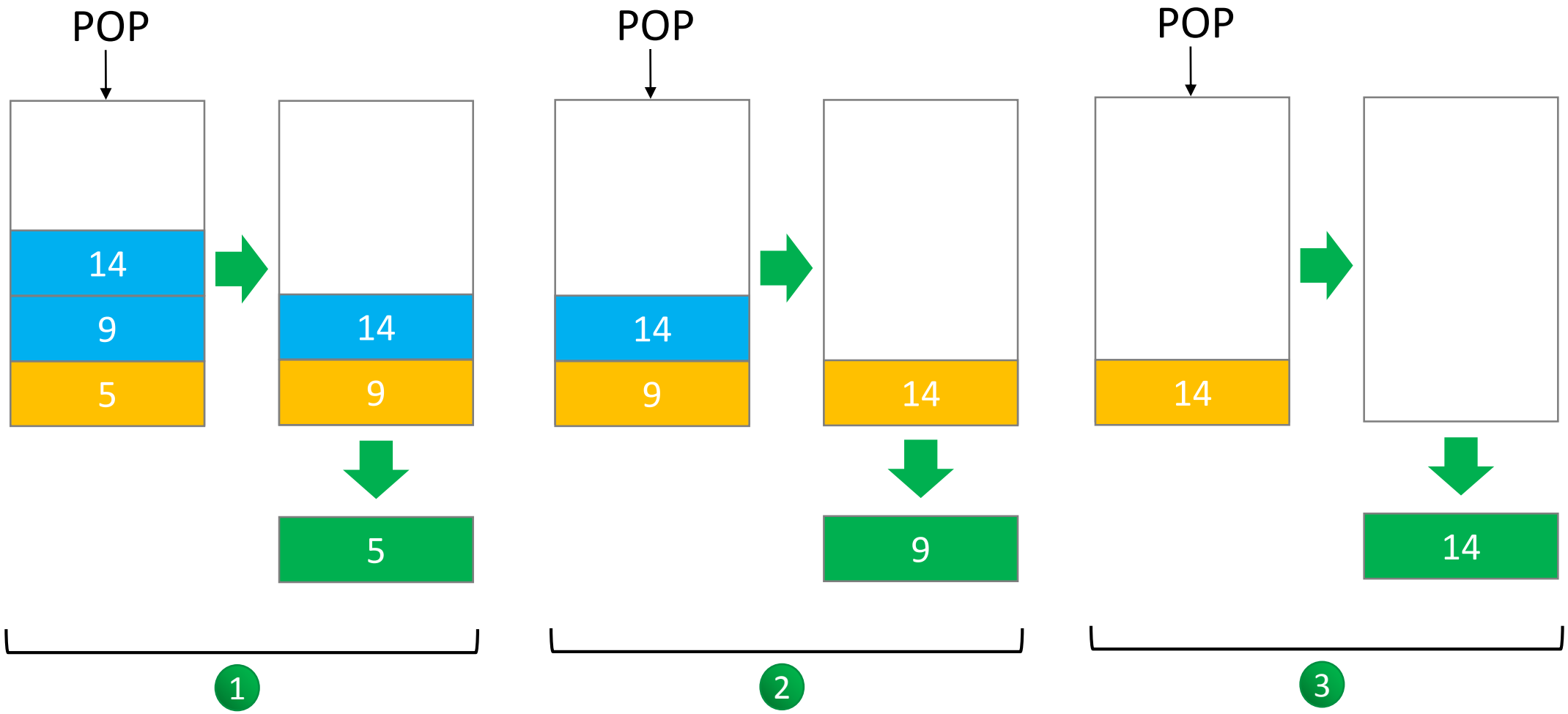
- Добавить/вставить (enqueuer/push)
- Извлечь (dequeuer/pop)
- Показать первый элемент очереди (front/peek)
- Размер стека (size)



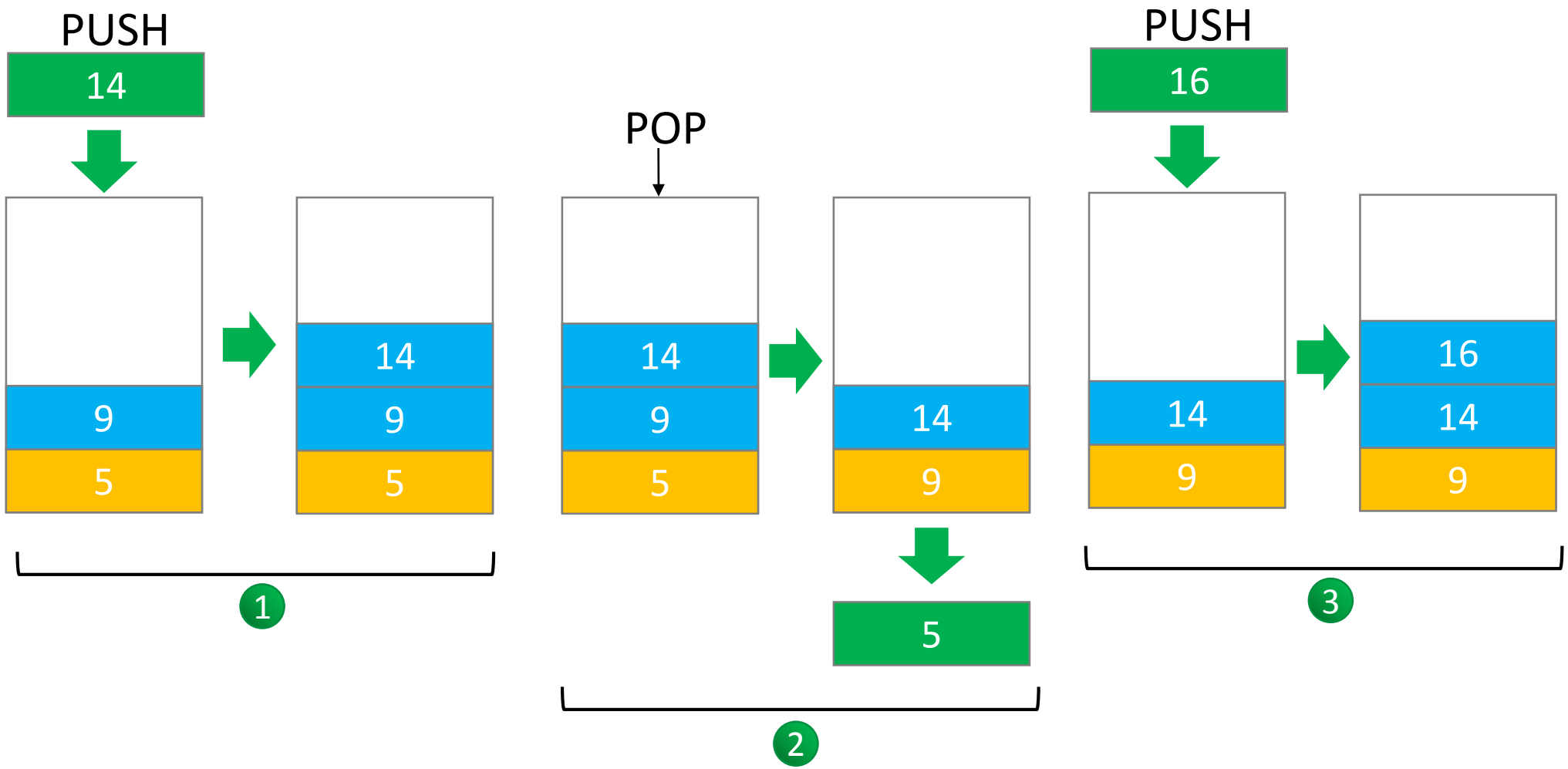
Вставка в очередь / PUSH



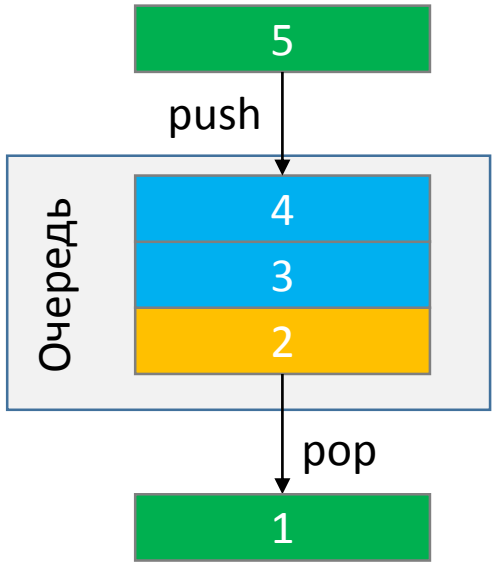
Извлечение из очереди / POP



PUSH/POP



Очередь

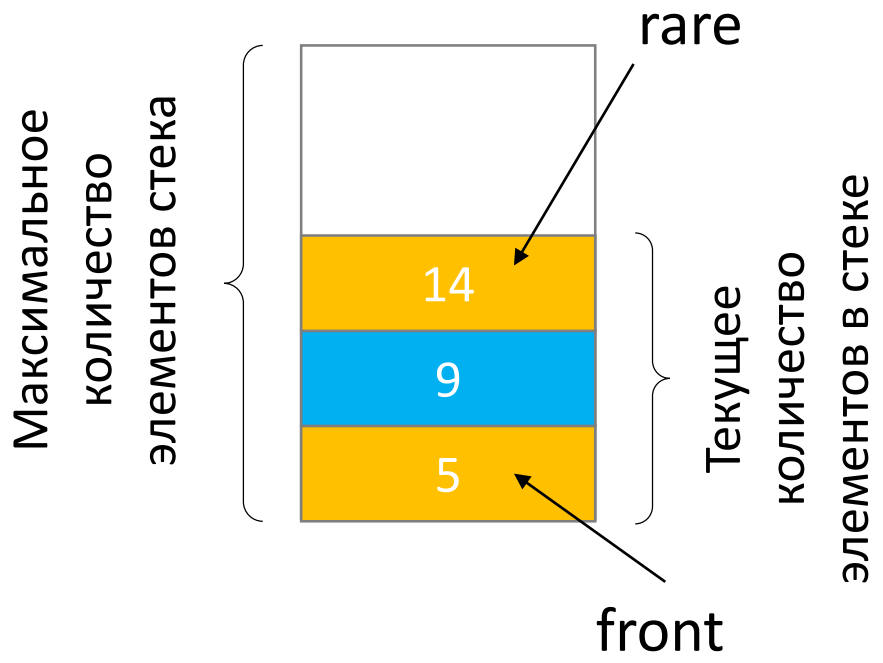


Дек

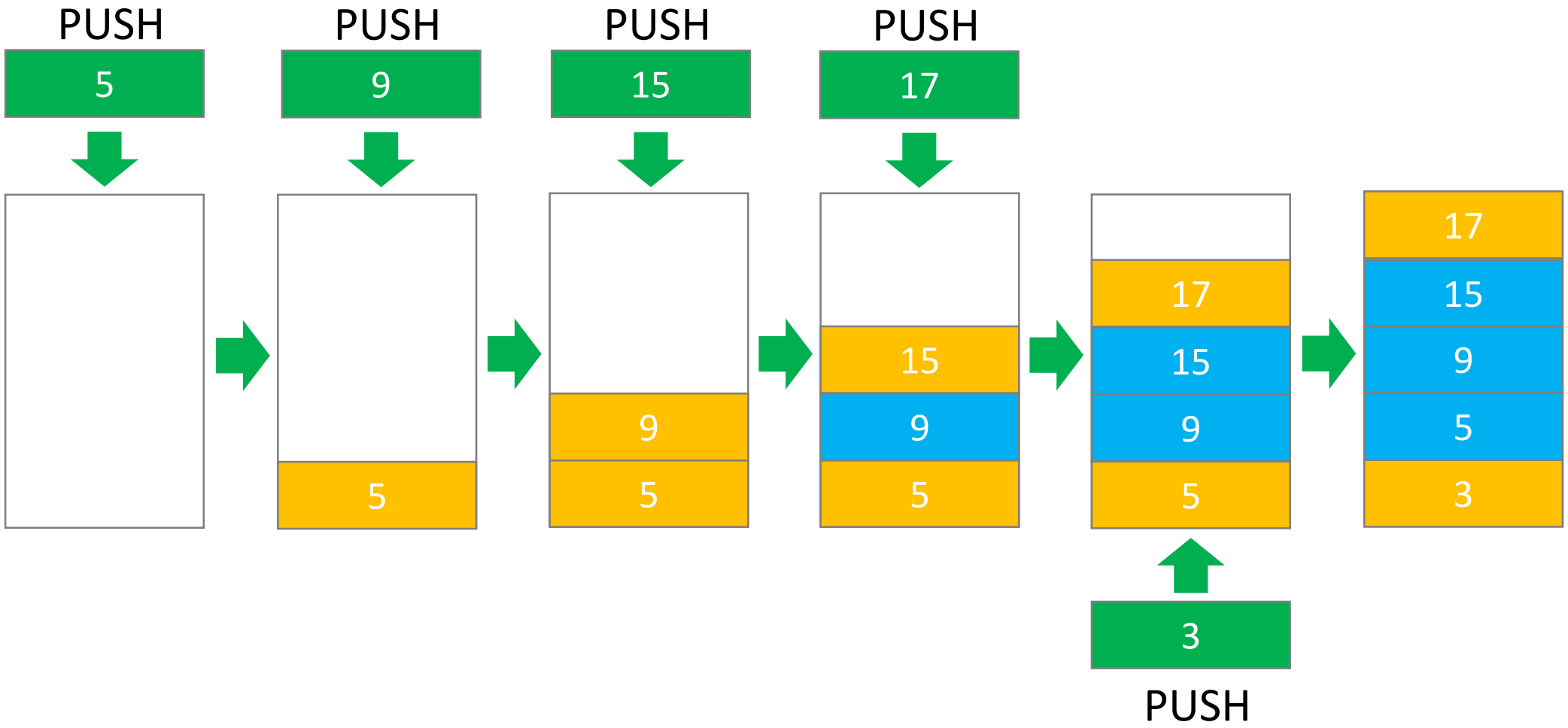
Дек (двухсторонняя очередь) – коллекция, элементы которой можно добавлять и извлекать как с начала, так и с конца.

Основные операции:

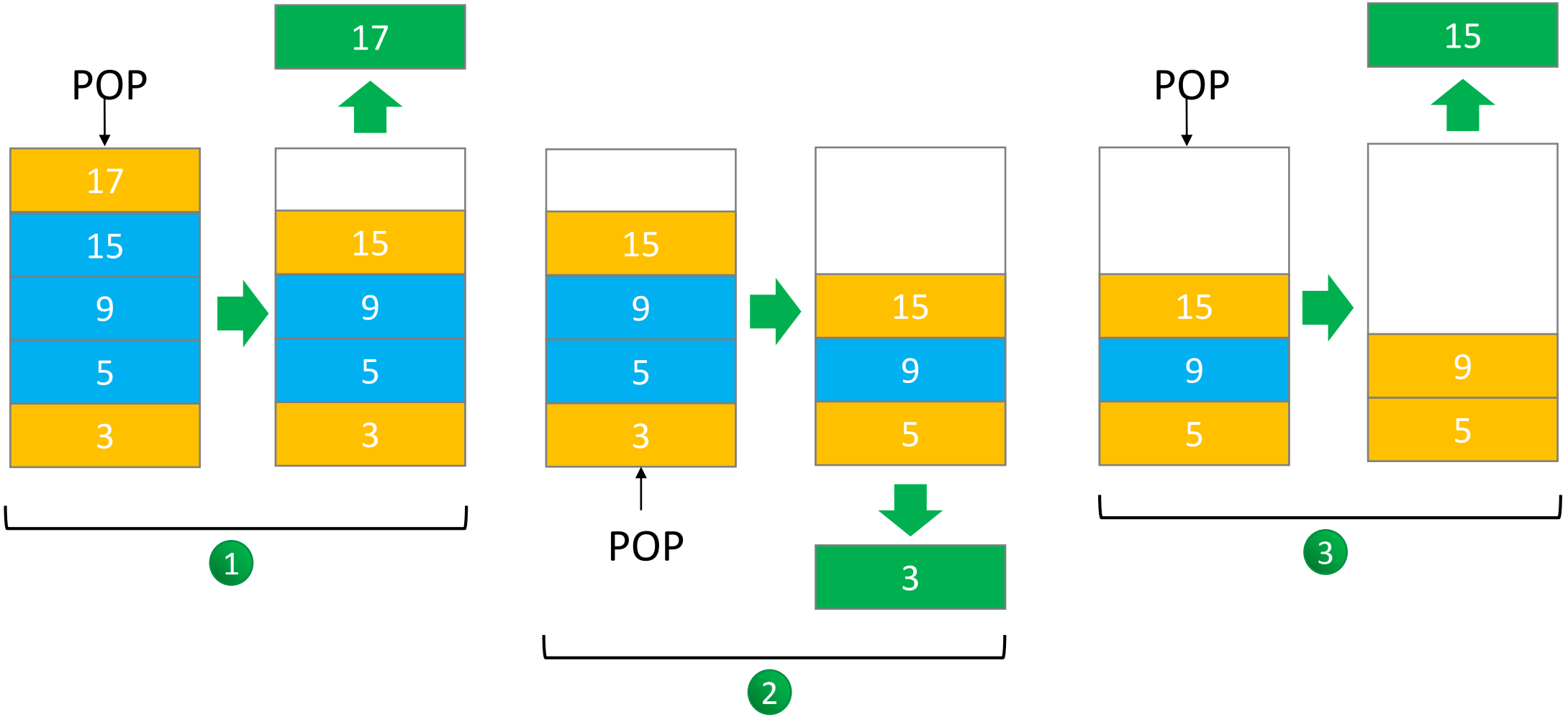
- Добавить/вставить (enqueuer/push)
  - сверху/сзади/в конец
  - снизу/спереди/в начало
- Извлечь (dequeuer/pop)
  - сверху/сзади/с конца
  - снизу/спереди/с начала
- Показать первый элемент дека (front)
- Показать последний элемент дека (back/rare)
- Размер стека (size)



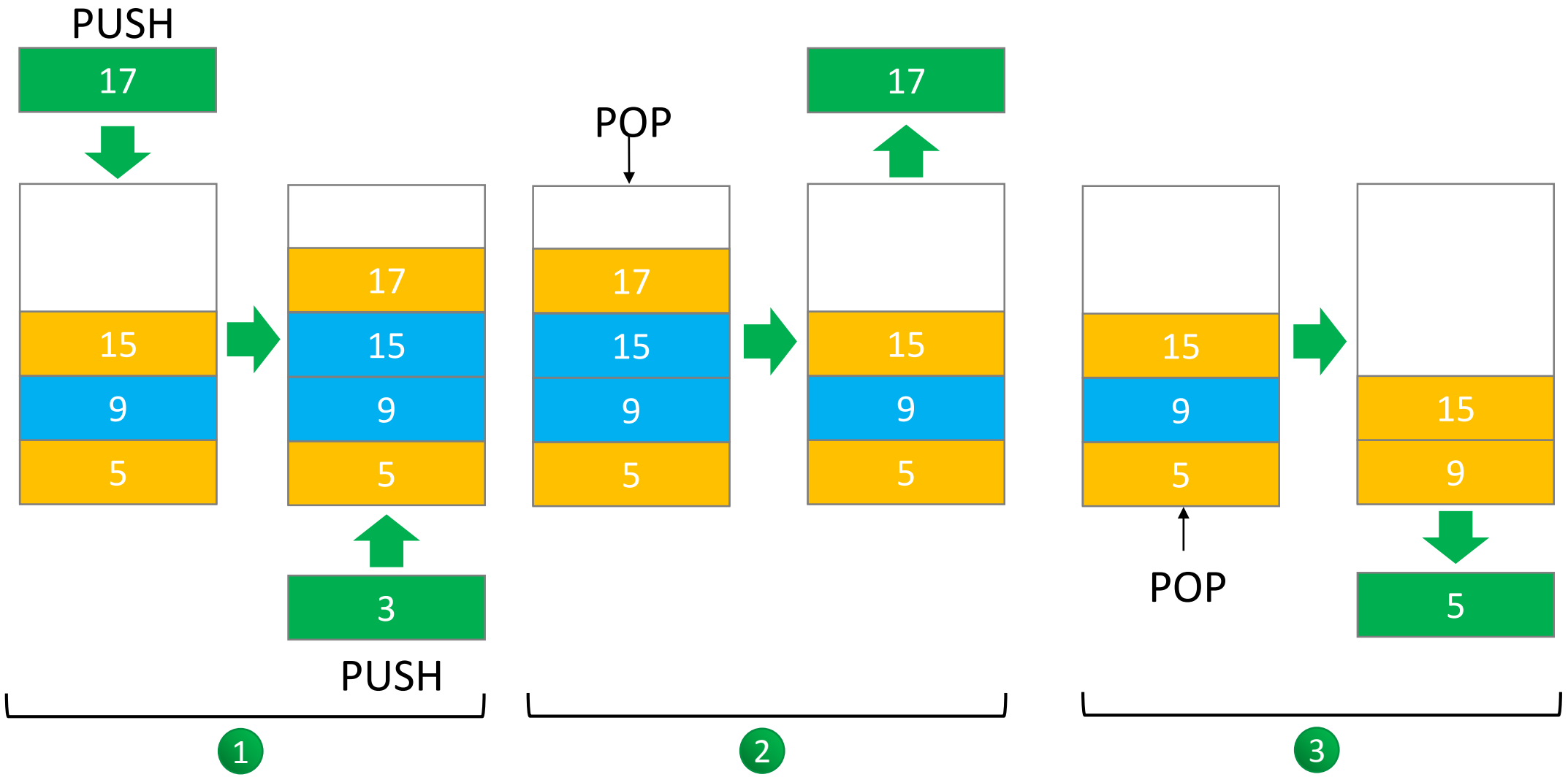
Вставка в края дека / PUSH



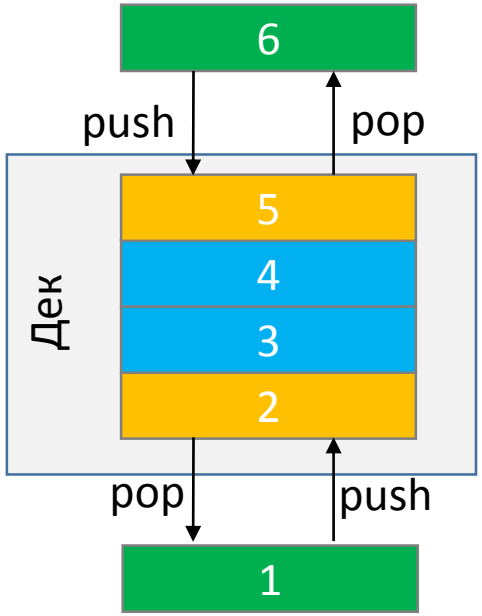
Извлечение из дека / POP



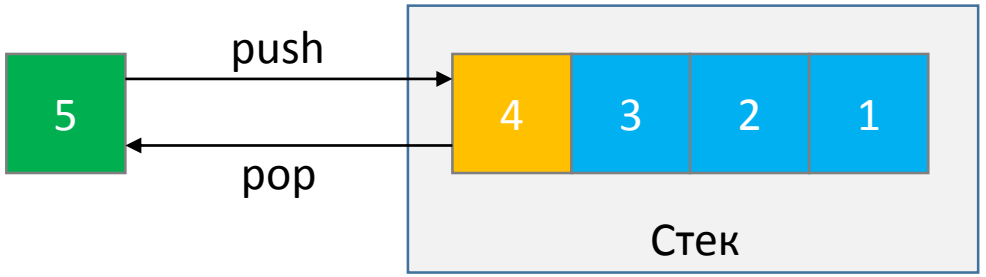
PUSH/POP


PUSH
17
15
9
5
17
15
9
5
3
PUSH
1
POP
17
15
9
5
17
15
9
5
2
15
9
5
POP
15
9
5
5
3

Дек



Стек. Очередь. Дек



Список в Python. Операции

Операция	Сложность
index []	O(1)
index assignment	O(1)
append	O(1)
pop()	O(1)
pop(i)	O(n)
insert(i,item)	O(n)
del operator	O(n)
iteration	O(n)
contains (in)	O(n)
get slice [x:y]	O(k)
del slice	O(n)
set slice	O(n+k)
reverse	O(n)
concatenate	O(k)
sort	O(n log n)
multiply	O(nk)

http://interactivepython.org/runestone/static/pythonds/AlgorithmAnalysis/Lists.html

Дек в Python. Операции

```
from collections import deque
```

Метод	Описание
append(x)	Добавить элемент с правой стороны
appendleft(x)	Добавить элемент с левой стороны
clear()	Удалить все элементы
count(x)	Посчитать количество элемента x с деке
extend(iterable)	Расширить правую часть дека за счет добавления элементов из iterable
extendleft(iterable)	Расширить левую часть дека за счет добавления элементов из iterable. При этом элементы добавляются в обратном порядке
pop()	Удалить и возвратить элемент из правой части дека. Если нет элементов, то вызывается IndexError
popleft()	Удалить и возвратить элемент из левой части дека. Если нет элементов, то вызывается IndexError
remove(value)	Удалить первый встретившийся элемент value. Если элемент не найден, то ValueError
reverse()	Реверсировать дек
rotate(n)	Сдвинуть элементы дека на n шагов вправо, если n – положительное целое число. Если n – отрицательное число, то дек сдвигается влево. d.rotate(1) == d.appendleft(d.pop())
maxlen	Максимальный размер дека

Indexed access is O(1) at both ends but slows to O(n) in the middle

https://docs.python.org/3.4/library/collections.html#collections.deque

https://official.contest.yandex.ru/contest/1864/enter/