

Алгоритмы и структуры данных на Python. Интерактивный  
курс

## Урок 6



# Работа с динамической памятью

Представление в памяти  
коллекций. Управление  
памятью.

# План

- Память компьютера.
- Проблемы управления памятью.
- Способы выделения и фазы управления памятью.
- Основные методы управления памятью.



# Основные фазы работы с памятью



# Проблемы управления памятью

Память не бесконечна

Механизм управления  
памятью:  
автоматический или  
ручной

Работа с общей  
памятью



# Способы выделения памяти

**СТАТИЧЕСКАЯ  
ИНФОРМАЦИЯ**

**ДИНАМИЧЕСКАЯ  
ИНФОРМАЦИЯ**



# Фазы управления памятью



# Основные методы управления памятью

- Статическое распределение памяти
- Стековое распределение памяти
- Представление памяти в виде кучи (heap)



# Статическое распределение памяти на примере двумерного массива

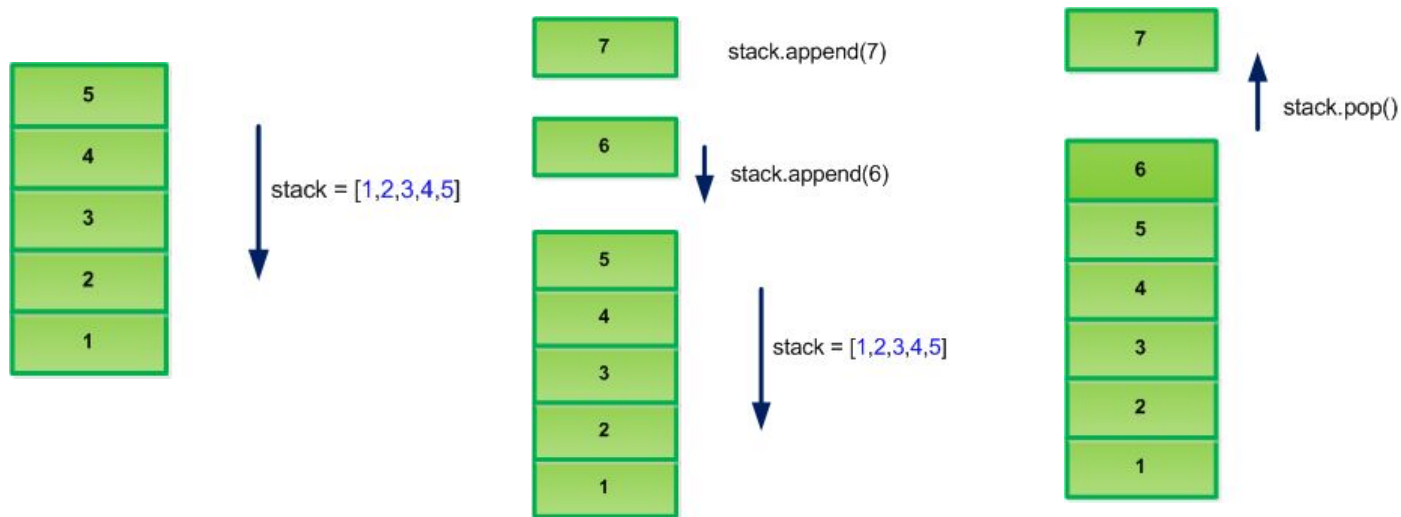


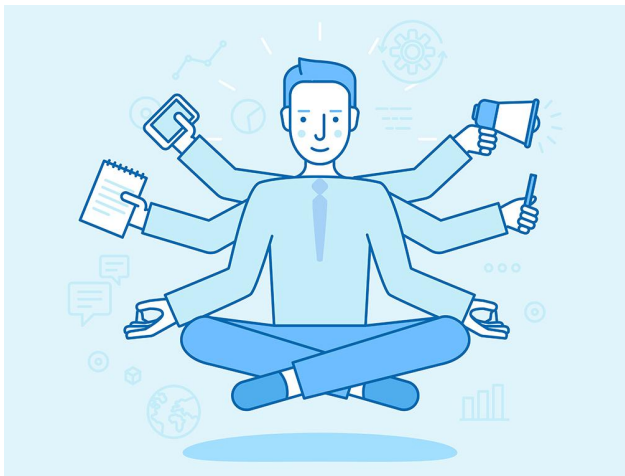
$$\text{Address} + \text{SizeElem} * \text{Cols} * i + \text{SizeElem} * j$$





# Стековое управление памятью





# Управление кучей



# Итоги:

## Теория

- Память компьютера.
- Проблемы управления памятью.
- Способы выделения и фазы управления памятью.
- Основные методы управления памятью.



# План

- Системы счисления
- Представление чисел в памяти (нули и единицы)



# Системы счисления

- Десятичная 42
- Двоична 0b101010
- Восьмеричная 0o52
- Шестнадцатеричная 0x2a



# Системы счисления

10	2	8	16
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7

10	2	8	16
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F



# Целые числа в двоичном коде

Прямой порядок	12	0	0	0	0	1	1	0	0
Обратный порядок	-12	1	1	1	1	0	0	1	1
Дополнительный порядок	-12	1	1	1	1	0	1	0	0



# Вещественные числа в двоичном коде

$$312,3125 == 100111000,0101$$

$$100111000,0101 == 1,001110000101 \times 2^8 == 1,001110000101 \times 10^{1000}$$





# Вещественные числа в двоичном коде

Название	Знак	Порядок									Мантисса						
Индекс	31	30	29	28	27	26	25	24	23		22	21	20	19	18	17	16
Бит	0	0	0	0	0	1	0	0	0	1	0	0	1	1	1	0	0

Название	Мантисса															
Индекс	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Бит	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0



# Итоги:

## Теория

- Системы счисления
- Представление чисел в памяти (нули и единицы)

## Практика

- Функции преобразования систем счисления



# План

- Список (`list`) - как он работает в недрах Python?

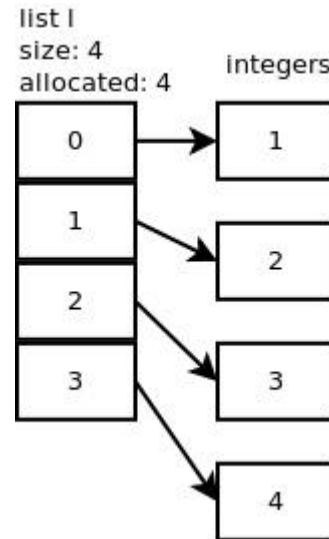
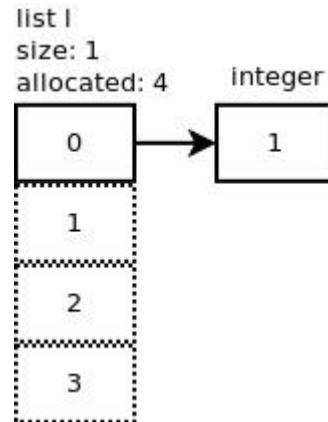


# Список. List

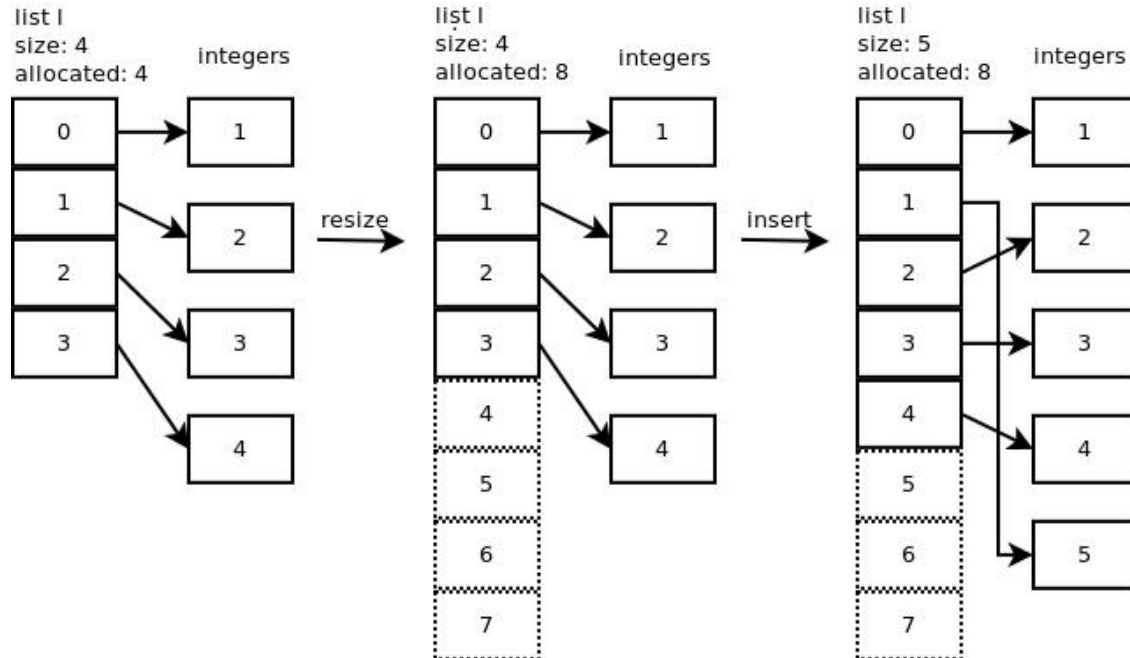
**Список (`list`)** - упорядоченная изменяемая коллекция объектов произвольных типов, к которым можно обратиться по индексу.



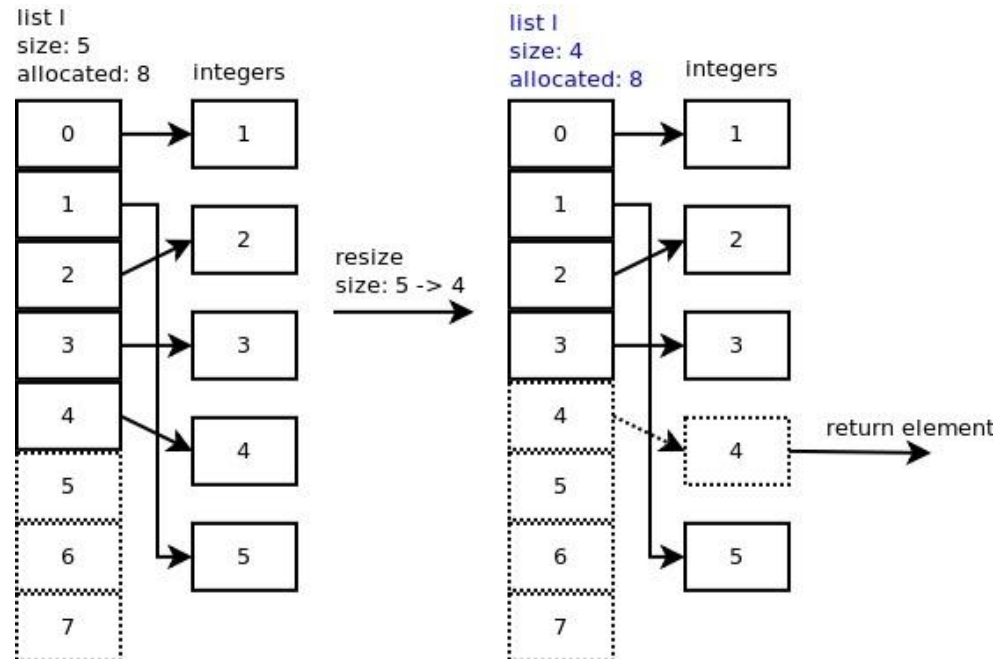
# list.append, list.extend



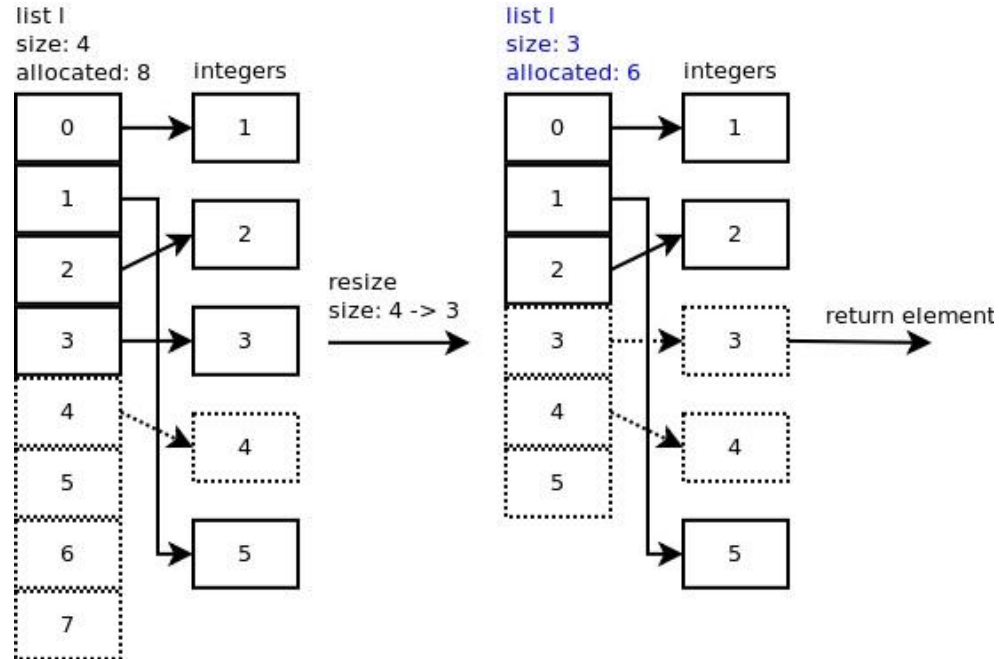
# list.insert



# list.pop

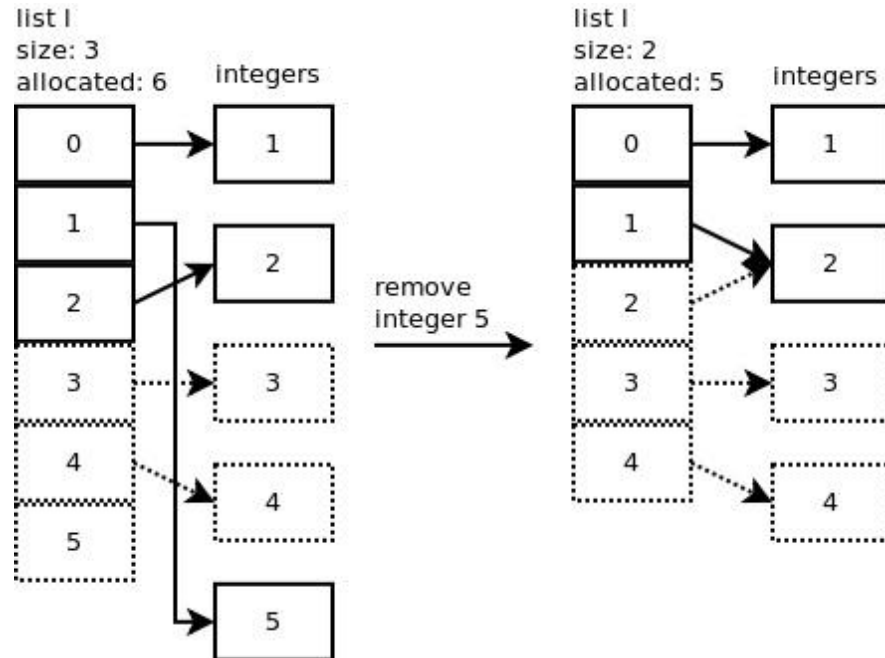


# list.pop





# list.remove



# Итоги:

## Теория и Практика

- Список (`list`) - основные методы и как они работают.



# План

- Сколько места занимают объекты в памяти?



# Итоги:

## Теория

- Сколько места занимают объекты в памяти

## Практика

- Программа для рекурсивного подсчёта занимаемой объектом памяти

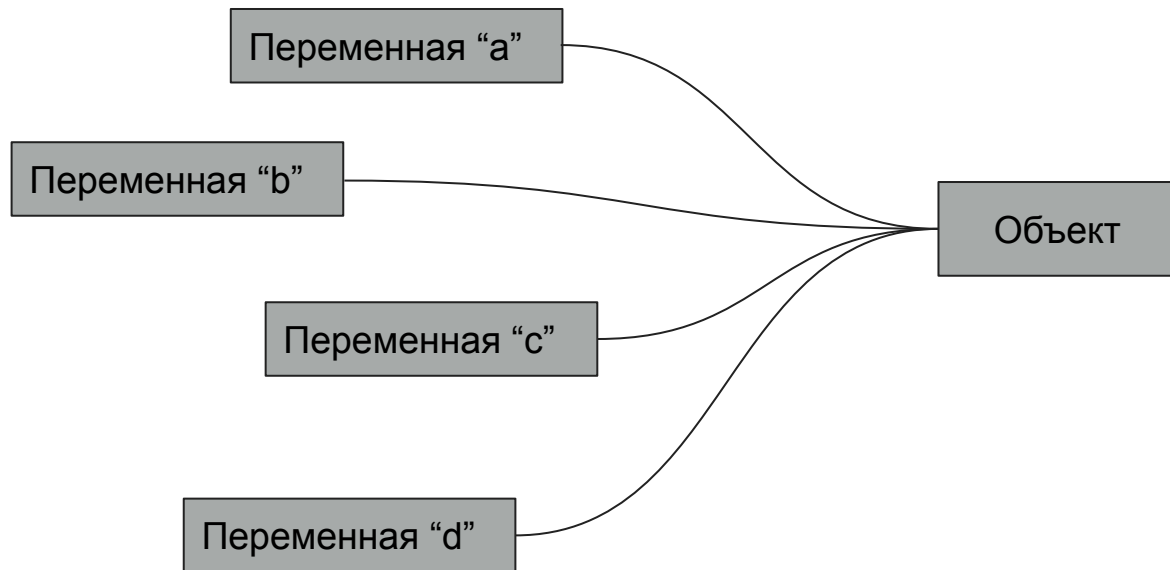


# План

- Структура объекта в памяти



# Переменная в Python



# Структура объекта в памяти

Счетчик ссылок	Указатель на тип	Содержимое
7	1624869344	42



# Расшифровка `struct.unpack`

Символ	Значение C	Значение Python	Длина в байтах
c	char	Строка из одного символа	1
i	int	int	4
l	long	int	4
L	unsigned long	int	4
d	double	float	8





# Итоги:

## Теория и Практика

- Структура объекта в памяти



# Домашнее задание

1. Подсчитать, сколько было выделено памяти под переменные в программах, разработанных на первых трех уроках.

Выберите 3 любые ваши программы для подсчёта.



# План

- Разбор домашнего задания

