

Python Basics v2.0

модуль I

Условные конструкции

Сергей Колесник

Минск, WGU, 2016

Что было в прошлый раз?

```
[::-1]  
python len  
lower str  
split  
\n startswith  
replace  
format
```

Элементарные типы данных

Числа

Строки

Логический тип

Пустой тип

Списки

Кортежи

Словари

Файлы

Множества

Логический тип

- Имеет два значения:
 - `True` («истина»)
 - `False` («ложь»)
- Задаем интерпретатору вопрос, в ответ получаем «да» или «нет»
 - `1 < 2 → True`
 - `len('hi!') > 5 → False`

Преобразования типов

- `type(True) → bool`

Преобразования типов

- `type(True) → bool`
- **Приведение к числам:**
 - `int(True) → 1`
 - `int(False) → 0`

Преобразования типов

- `type(True) → bool`
- **Приведение к числам:**
 - `int(True) → 1`
 - `int(False) → 0`
- **Приведение к логическому типу:**
 - `bool(0) → False`
 - `bool(2) → True`
 - `bool('hello') → True`
 - `bool('') → False`
 - `bool() → False`

Пустое значение объекта
в рамках его типа означает
`False`

Все остальные значения
означают `True`

Операции, возвращающие bool

- `<, >, <=, >=`
- `!=, ==`
- `in, not in`
- `not, and, or`
- `is, is not`



Операции, возвращающие bool

- `<, >, <=, >=`
- `!=, ==`
- `in, not in`
- **`not, and, or`**
- **`is, is not`**



is

- Проверка идентичности двух объектов

`1 is 2 → False`

`1 is 1 → True`

is

- Проверка идентичности двух объектов

`1 is 2 → False`

`1 is 1 → True`

`1 == 2 → False`

`1 == 1 → True`



равенство ??

is

- Проверка идентичности двух объектов

```
x = 300
```

```
y = 300
```

```
x == y → True
```



равны

is

- Проверка идентичности двух объектов

```
x = 300
```

```
y = 300
```

```
x == y → True
```

```
x is y → False
```



равны



но не идентичны!

is

- Проверка идентичности двух объектов

x = 300

y = x

is

- Проверка идентичности двух объектов

```
x = 300
```

```
y = x
```

```
x == y → True
```

```
x is y → True
```

is not

- противоположный is

```
x = 300
```

```
y = 'abc'
```

```
x is not y → True
```

Практика

Проверка отсутствия во входной строке слова «ISIS»

Input: Строка

Output: False, если слово есть в вводе, иначе True

Практика

Проверка того, что ввод не начинается с фразы «White House»

Input: Строка

Output: False, если начинается с фразы, иначе True

Сравнение с bool

- Сравнивать следует с помощью оператора `is`

`x is True`



`x == True`



Сравнение с bool

- Почему не подходит оператор равенства?

```
x == True
```

```
2 == True → False
```

```
1 == True → True
```

Сравнение с bool

- Почему не подходит оператор равенства?

```
x == True
```

```
2 == True → False
```

```
1 == True → True
```

В этих двух случаях значение не совпадает с True, однако результат разный

not

- Превращает bool-value в противоположенное

```
not '123'.isdigit() → False
```


and

- Требуется «выполнения» двух bool-value

```
x.isdigit() and x.endswith('abc')
```

→ False

or

- Требуется «выполнения» одного bool-value

```
x.isdigit() or x.endswith('abc')
```

→ False или True (зависит от x)

Элементарные типы данных

Числа

Строки

Логический тип

Пустой тип

Списки

Кортежи

Словари

Файлы

Множества

«Пустой» тип

- Существует единственный объект этого типа: `None`
- Смысл: показать отсутствие значения

«Пустой» тип

- Существует единственный объект этого типа: `None`
- Смысл: показать отсутствие значения
`age = get_user_age()`

Сравнение с None

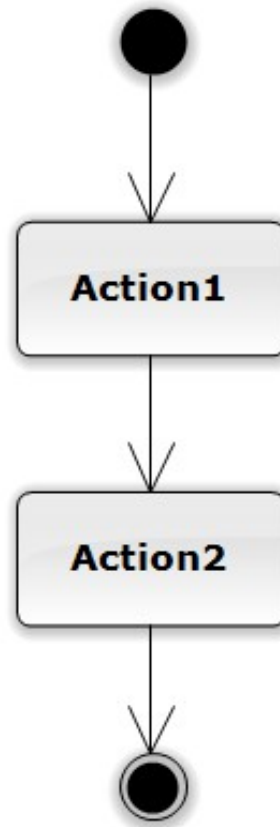
- Сравнивать следует с помощью оператора `is`

`age is not None`

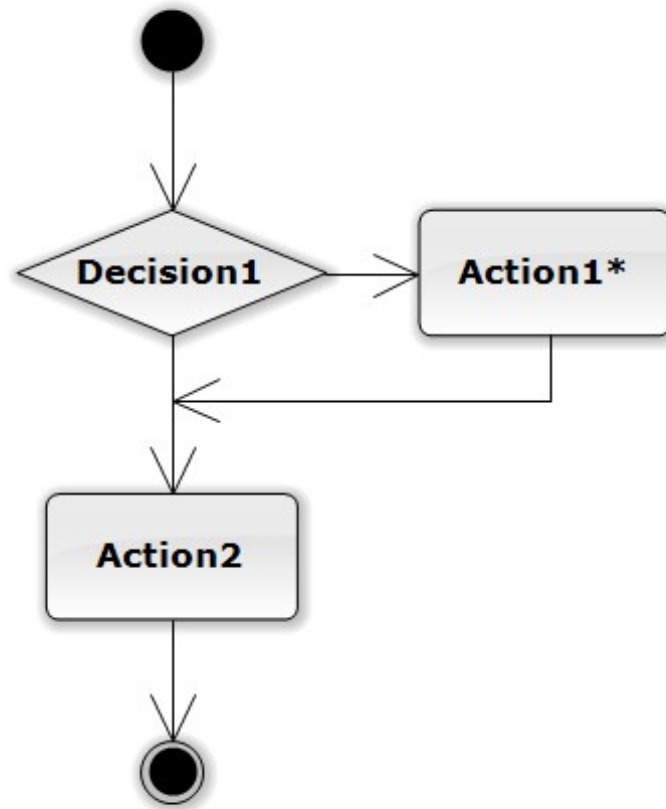
- `None < -10 ** 1000000000 → True`

Условные конструкции

Линейная структура



Ветвящаяся структура



Ветвящаяся структура

```
if <bool-expression>:  
    <action1*>  
<action2>
```

Пример

Выбор черного или белого цвета с проверкой корректности ввода.

Input: Строка «black» или «white».

Output: Строка «<color> is nice choice!».

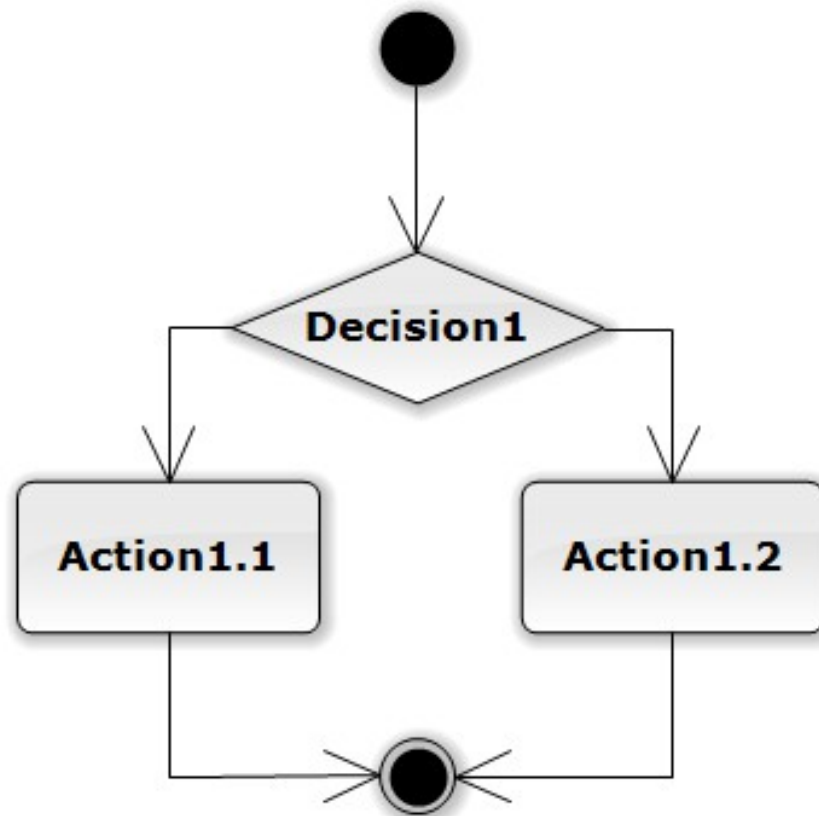
Практика

Вычисление года рождения пользователя на основе его возраста с проверкой корректности ввода.

Input: Целое число — текущий возраст пользователя.

Output: Целое число — год рождения, без учета дней и месяцев.

Ветвящаяся структура с двумя альтернативами



Ветвящаяся структура с двумя альтернативами

```
if <bool-expression>:  
    <action1.1>  
else:  
    <action1.2>
```

Практика

Проверка делимости числа на 10.

Input: Целое число

Output: Yes, если введенное число делится на 10, иначе No

Практика

Определение максимального из двух чисел.

Input: Два целых разных числа

Output: Номер того числа, которое больше: 1 или 2

Практика

Определение максимального из двух чисел.

Input: Два целых числа

Output: Номер того числа, которое больше: 1 или 2, либо =, если числа одинаковые.

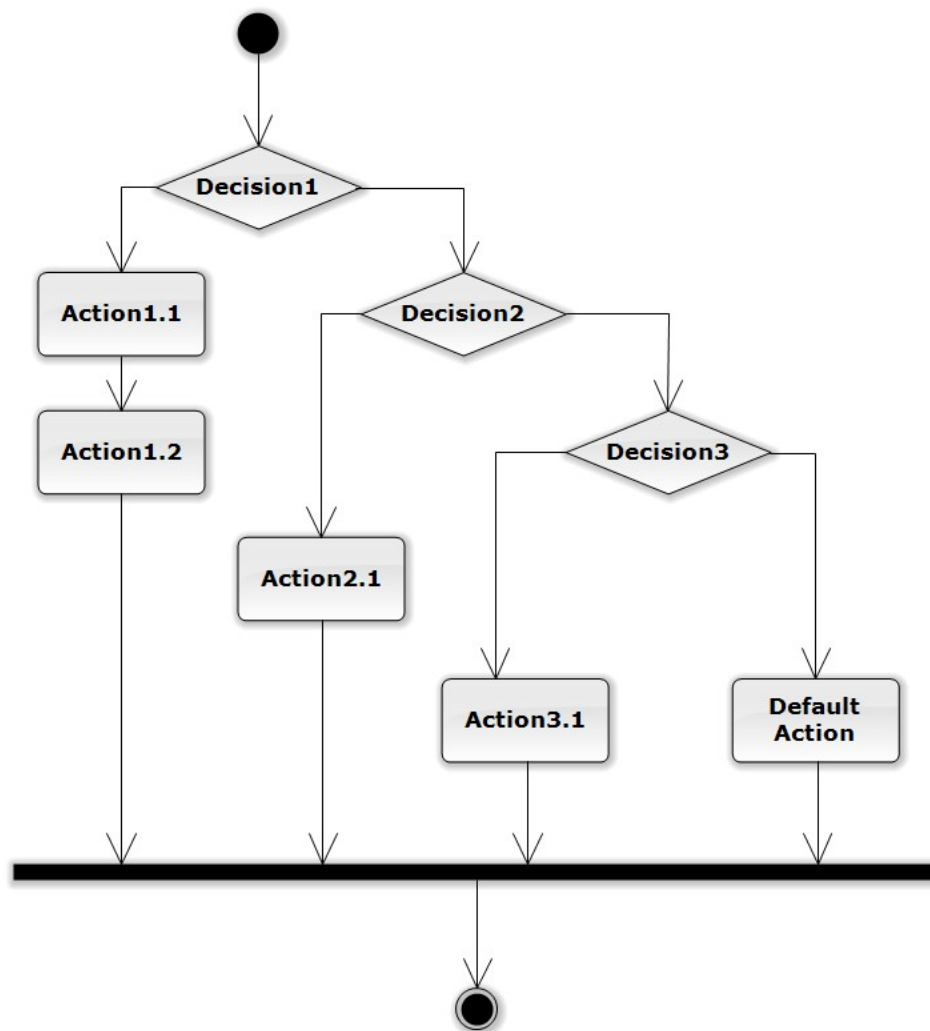
Практика

Определение максимального из трех чисел.

Input: Три целых разных числа

Output: Номер того числа, которое больше: 1, 2 или 3

Комбинированная структура



Комбинированная структура

```
if <bool-expression1>:  
    <action1.1>  
    <action1.2>  
elif <bool-expression2>:  
    <action2.1>  
elif <bool-expression3>:  
    <action3.1>  
else:  
    <default-action>
```

Практика

Определение названия дня недели.

Input: Номер дня недели

Output: Название этого дня, либо сообщение о некорректном номере.

Практика

Проверка, одного ли знака, числа.

Input: Два целых ненулевых числа

Output:

- Yes, если оба числа больше или меньше нуля
- No, если одно больше нуля, а второе меньше

Практика

Проверка, одного ли знака, числа.

Input: Два целых числа

Output:

- Yes, если оба числа $>$ или $<$ нуля
- No, если одно $>$ нуля, а второе — $<$
- Zero, если одно из чисел $==$ нулю
- Both zero, если оба — нули

Практика

Определение четности двух чисел.

Input: Два целых числа

Output:

- odd, если оба нечетные
- even, если оба четные
- even-odd, если разные

Практика

Определение номера квадранта, в котором расположена 2D-точка.

Input: Два вещественных ненулевых числа

Output: I, II, III или IV

Практика

Определение номера квадранта, в котором расположена 2D-точка.

Input: Два вещественных числа — координаты точки

Output: I, II, III, IV, Ох, Оу или О

Резюме

- Условные конструкции:
 - Проверка корректности входных данных
 - Нелинейный поток выполнения программы

Спасибо за внимание