

ТИПЫ ДАННЫХ



СГУГиТ
СИБИРСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГЕОСИСТЕМ И ТЕХНОЛОГИЙ

Р. В. Гришин
mail-to: r.grishin54@gmail.com

СОДЕРЖАНИЕ ЛЕКЦИИ

- ☐ Примитивные типы данных
- ☐ Строки
- ☐ Определяемые пользователем порядковые типы
- ☐ Массивы
- ☐ Ассоциативные массивы
- ☐ Списки, кортежи
- ☐ Объединения
- ☐ Указатели и Ссылки



ВВЕДЕНИЕ

- ❑ Каждая программа использует данные, **явно** или **неявно**, чтобы получить результат.
- ❑ Все программы определяют набор операций, которые должны быть применены к определенным данным в определенной последовательности.
- ❑ Данные в самых примитивных формах внутри компьютера - это просто набор битов.
- ❑ Основные различия между языками существуют в типах разрешенных данных, в типах доступных операций и в механизме, предусмотренном для управления последовательностью, в которой операции применяются к данным.
- ❑ Большинство языков программирования предоставляют набор простых сущностей данных, а также механизм для построения новых сущностей данных из них.



ТИП ДАННЫХ

Тип данных - это набор значений вместе с набором операций над теми значениями, которые имеют определенные свойства.

Каждый язык поставляется с набором предопределенных типов, из которых построены все другие типы.

Основные элементы спецификации типа данных:

- ☐ Атрибуты, которые различают объекты и типы данных.
- ☐ Значения, которые могут иметь объекты данных этого типа;
- ☐ Операции, которые определили возможные манипуляции с объектами данных этого типа.



ИНФОРМАЦИЯ О ТИПЕ

- ❑ Данные программы могут быть классифицированы в соответствии с их типами.
- ❑ **Неявная** информация о типах включает в себя типы констант и значений, типы, которые могут быть выведены из соглашения об именах, и типы могут быть выведены из контекста.
 - ❖ Пример: число 2 неявно является целым числом в большинстве языков
 - ❖ Пример: TRUE является логическим
- ❑ **Явная** информация о типе в основном содержится в объявлении переменных.
 - ❖ Пример: `var x: array[1..10] of integer;`
 - ❖ Пример: `var b: boolean;`



ПРИМИТИВНЫЕ ТИПЫ ДАННЫХ

- ❑ Algol-подобные языки (Pascal, Algol68, C, Modula-2, Ada, C++), все классифицируют типы в соответствии с базовой схемой, с незначительными вариациями.
- ❑ Примитивные типы также называются базовыми типами или скалярными типами или неструктурированными типами или элементарными типами.
- ❑ Скалярный тип - это тип, элементы которого состоят из неделимых сущностей (единое значение или атрибут данных).



ЦИСЛОВЫЕ ТИПЫ ДАННЫХ

Многие ранние языки программирования имели только числовые примитивные типы. Числовые типы по-прежнему играют центральную роль среди коллекций типов, поддерживаемых современными языками.

- ☐ Integer
- ☐ Float и Double
- ☐ Decimal



INTEGERS

Набор целочисленных значений, определенных для типа, образует упорядоченное подмножество, в пределах некоторых конечных границ, бесконечного набора целых чисел.

Операции с integers объектами данных включают:

- ☐ Арифметические операции
 - ❖ Бинарные арифметические операции (+, -, *, /);
 - ❖ Унарные арифметические операции (+, -);
- ☐ Операции сравнения (>, >=, <, <=, ==)
- ☐ Битовые операции (&, |, <<, >>)



FLOATING-POINT

Типы данных с плавающей точкой моделируют реальные числа, но для многих реальных значений их представления являются лишь приближенными.

Эта модель эмулирует научную нотацию где число N может быть выражено как $N = m \cdot 2^k$

Числа состоят из 3 полей:

- ❑ Бит знака (1 – отрицательное, 0 – положительное)
- ❑ Значения (8 бит) варьируются от 0 до 355, что соответствует экспоненте 2, которая находится в диапазоне от -127 до 128;
- ❑ мантисса из 23 бит. Первое число всегда равно 1, затем оно автоматически вставляется аппаратурой, обеспечивая дополнительный 24-й бит точности.

Sign bit	Exponent	Mantissa
-------------	----------	----------

$$12.345 = \underbrace{12345}_{\text{significand}} \times \underbrace{10^{-3}}_{\text{base}}^{\text{exponent}}$$



СГУГиТ

СИБИРСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГЕОСИСТЕМ И ТЕХНОЛОГИЙ

BOOLEAN

Объекты данных, имеющие одно из двух значений (true или false).

Наиболее распространенные операции с логическими типами включают в себя присваивание и следующие логические операции:

- ☐ **And (&&)**
- ☐ **Or (||)**
- ☐ **Not (!)**

Boolean - один из основных типов данных в программировании, который широко применяется в различных контекстах, начиная от управления потоком программы до логических проверок и хранения состояний.



СТРОКИ

Строка - это тип данных, предназначенный для хранения текстовой информации, состоящей из последовательности символов. Строка может содержать буквы, цифры, символы пунктуации и другие символы. Она может быть пустой (не содержащей символов) или иметь различную длину.

Операции поддерживаемые строками:

- ☐ Присваивание и копирование
- ☐ Сравнение (=, > и т. д.)
- ☐ Конкатенация
- ☐ Длина
- ☐ Ссылка на подстроку
- ☐ Сопоставление с образцом



ENUMERATION

Тип перечисления - это тип, в котором все возможные значения, являющиеся именованными константами, представленными или перечисленными в определении. Перечисления обеспечивают способ определения и группировки коллекций именованных констант, которые называются константами перечисления.

Пример:

- ❑ C#: `enum days {mon, tue, wed, thu, fri, sat, sun};`
- ❑ C++: `enum colors {red, blue, green, yellow, black};`
`colors myColor = blue; yourColor = red;`



МАССИВЫ

Массив представляет собой однородный совокупность элементов данных, в котором отдельный элемент идентифицируется по его положению в совокупности относительно первого элемента.

Инициализация массивов:

- ❑ C/C++: `int list [] = {1, 3, 5, 6};`
`char *names [] = {"Mike", "Fred", "Mary Lou"};`
- ❑ Ada: `List : array (1..5) of Integer := (1 => 17, 3 => 34, others => 0);`
- ❑ Java: `String[] names = ["Bob", "Jake", "Darcie"] ;`



ТИПЫ МАССИВОВ

- ❑ **Статический массив** - это массив, в котором диапазоны статически связаны, а распределение памяти выполняется до времени выполнения.
- ❑ **Динамический массив с фиксированным стеком** - это массив, в котором диапазоны статически связаны, но распределение происходит во время обработки объявления в процессе выполнения.
- ❑ **Стеково-динамический массив** - это массив, в котором диапазоны и распределение хранилища динамически привязываются во время работы.
- ❑ **Фиксированный динамический массив кучи** похож на фиксированный динамический массив стека тем, что диапазоны и привязка к хранилищу фиксируются после выделения хранилища.
- ❑ **Динамический массив в куче** - это массив, в котором привязка диапазонов и распределение хранилища являются динамическими и могут меняться любое количество раз за время существования массива.



АССОЦИАТИВНЫЕ МАССИВЫ

Ассоциативный массив - это неупорядоченный набор элементов данных, которые индексируются равным количеством значений, называемых ключами.

Примеры:

- ❑ Perl: `%salaries = ("Gary" => 75000, "Perry" => 57000, "Mary" => 55750);`
`$salaries{"Perry"} = 58850;`
- ❑ Python: `salaries = {"Gary": 75000, "Perry": 57000, "Mary": 55750}`
`salaries["Perry"] = 59000`



СПИСКИ

Список (list) - это структура данных в программировании, представляющая собой упорядоченный набор элементов. В списке могут храниться элементы любых типов данных (целые числа, строки, другие списки и т. д.), и они могут быть изменяемыми в большинстве языков программирования.

Список (list)

```
lst = []
```

```
lst = [1, 2, 3]
```

```
lst1 = list(lst)
```

```
lst.append(4)
```

Swap

```
a, b = b, a
```

Распаковка

```
lst = [1, 2, 3, 4, 5]
```

```
a, b, *c = lst # 1, 2, [3, 4, 5]
```

```
a, *b, c = lst # 1, [2, 3, 4], 5
```



СГУГиТ

СИБИРСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГЕОСИСТЕМ И ТЕХНОЛОГИЙ

КОРТЕЖИ

Кортеж (tuple) - это структура данных в программировании, которая представляет собой упорядоченную коллекцию элементов различных типов. Основное отличие кортежа от списка заключается в его неизменяемости после создания.

Кортеж (tuple)

```
tup = ()
```

```
tup = (1,)
```

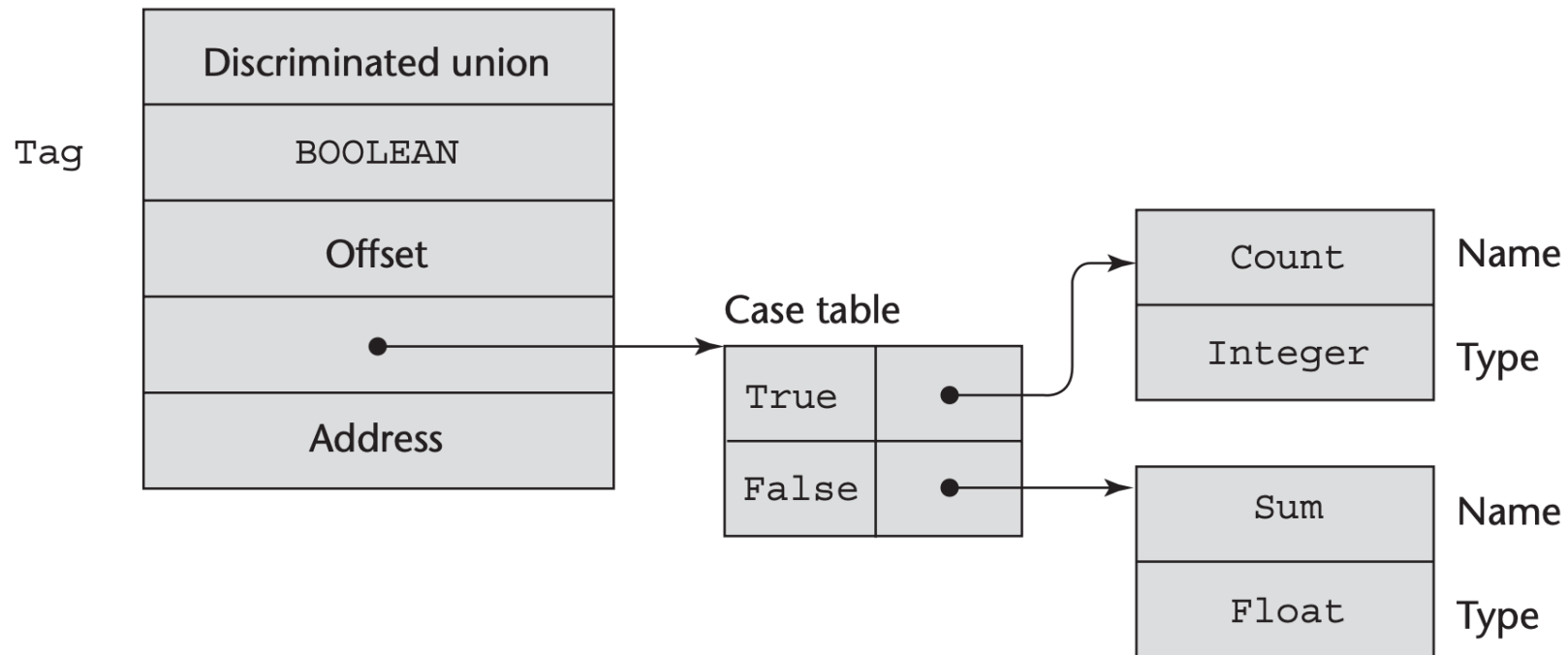
```
tup = 1, 2, 3
```

```
tup = tuple(lst)
```



UNIONS

Объединение - это тип, переменные которого могут хранить значения разных типов в разные моменты времени во время выполнения программы.



UNIONS

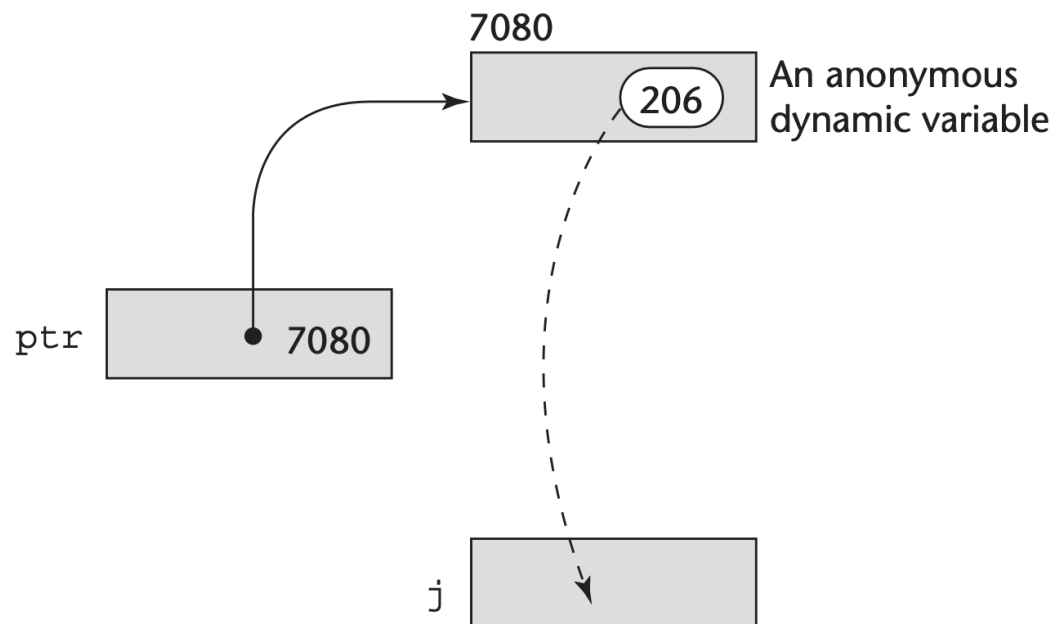
```
union flexType {  
    int intEl;  
    float floatEl;  
};  
union flexType el1;  
float x;  
...  
el1.intEl = 27;  
x = el1.floatEl;
```



УКАЗАТЕЛИ И ССЫЛКИ

Указатель - это тип, в котором переменные имеют диапазон значений, состоящий из адресов памяти и специального значения `nil`. Значение `nil` не является действительным адресом и используется для указания на то, что указатель в данный момент не может быть использован для ссылки на ячейку памяти.

`j = *ptr`



СГУГиТ

СИБИРСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГЕОСИСТЕМ И ТЕХНОЛОГИЙ

«**Типы данных** – это слова, из которых строиться великолепная симфония программирования»



СГУГиТ

СИБИРСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГЕОСИСТЕМ И ТЕХНОЛОГИЙ