

МатАнал

Забродин Денис Александрович

4 октября 2022 г.

Содержание

1	Определители	2
1.1	Тип объекта	2
1.2	Литералы и суффиксы	3
1.3	Явное и неявное преобразования	4
1.4	Атрибуты объектов	4
1.5	Область действия	5

1 Определители

1.1 Тип объекта

С машинной точки зрения тип объекта - это форма представления его значений в памяти компьютера и определяемый этой формой способ доступа к объекту и его частям.

С точки зрения программиста на ЯП высокого уровня это абстракция данных и операций над этими данными.

Объект данных - один или несколько элементов данных, рассматриваемый программой как одно простое или составное значение.

Любой объект данных характеризуется тремя основными свойствами

- объект данных связаны с определенным типом
- место хранения связанной с объектом информации (в ЯП высокого уровня на объекты данных ссылаются с помощью определенных пользователем имен, а не конкретных адресов памяти)
- объекты данных могут принимать конкретные значения.

Переменная определяется тремя характеристиками:

- **имена**, под которым переменная используется в программе
- **ссылкой**, однозначно идентифицирующей связанный с переменной объект данных в памяти компьютера и описывающей тип этого объекта
- **значением**, которое обрабатывает программа; значение в ходе выполнения программы может изменяться

Переменные вводятся с помощью определений, а ее значение задается в результате **инициализации** или **операции присваивания**

$$inta = 5;$$

Константа - конструкция языка программирования, значение которой не может изменяться в ходе выполнения программы.

$$\#define A5$$

Именованная константа - конструкция языка программирования, аналогичная переменной, с неизменяющейся в ходе выполнения программы значением, однако имеет имя и значение в памяти. `const int i = 5`

Тип данных определяет множество значений, которые могут принимать объекты этого типа, способ представления в памяти компьютера и множество операций, которые могут

1. Статическая типизация

- Слабая типизация
- Сильная
- Защитная

2. Динамическая типизация

Основные типы данных:

- простые
- перечисления
- составные
- тип-диапазон

Числовые типы подразделяются на:

- целые - множество целочисленных дискретных значений
- вещественные - представление с конечной точностью

Спецификаторы:

- short: короткий формат
- long: длинный формат

Квалификаторы:

- signed: знаковый (по умолчанию)
- unsigned: беззнаковый

По умолчанию целочисленные константы имеют тип signed int, их значения задаются целыми литералами в десятичном, восьмеричном

1.2 Литералы и суффиксы

Литералы

- 0b* - двоичное представление
- 0* - восьмеричное представление
- 0x* - шестнадцатеричное представление

Суффиксы:

- l(L) - число относится к типу long

- $u(U)$ - число беззнаковое

Допускается комбинация суффиксов: 42ul, 042ull

Для C/C++ предельные для различных типов данных представлены в заголовочном файле limits.h

Формат литералов: вещественный или экспоненциальный

Возможные суффиксы: l(L) (long) и f(F) (float)

Логический тип подразумевает два возможных значения: ЛОЖЬ и ИСТИНА

В языке программирования C++ существует тип данных bool, а в C - _Bool. Литералы в C++ задаются ключевыми словами false и true, а в C - 0 и 1

Формат: enum[имя типа перечисления]список констант[имя переменной]

Если в списке нет инициализаторов, то значения начинаются с 0 и увеличиваются на 1. Именованная константа с инициализатором получает заданное значение, а все следующие за ней именованные константы без инициализатора получают значения, последовательно увеличенные на 1 каждая

Иерархия типов C/C++: bool, char, short → int → unsigned int → long → unsigned long → long long → unsigned long long → float → double

Исходя из иерархии также выделяют:

- Сужающее преобразование
- Расширяющее преобразование

1.3 Явное и неявное преобразования

При явном преобразовании программист указывает тип, к которому необходимо преобразовать исходную переменную или константу.

1.4 Атрибуты объектов

Одно из основных понятий C - понятие объекта

Класс памяти определяет продолжительность существования объекта (время жизни) и способ размещения в памяти.

Способы хранения данных в памяти в C:

- **автоматическая** память
- **статическая** память
- **динамическая** память(heap)

- Начиная с C++11 **потоковая память**

Спецификаторы классов памяти:

- `auto` - автоматически выделяемая локальная память (в стеке). Время жизни - в пределах блока. Для C++14 слово `auto` имеет другой смысл - автоматическое выведение типа
- `register` автоматически выделяемая память, по возможности, в регистре процессора
- `static` память выделяется в статическом хранилище (сегмент данных), инициализируется один раз
- `extern` статическая внешняя переменная (определена в другом месте программы)

1.5 Область действия

2 Указатели и массивы

Указатель - переменная, значением которой служит адрес объекта конкретного типа или адрес в сегменте кода, по которому располагается исполняемый код функции.

Для получения адреса применяется унарная операция взятия адреса "&"