

# Многофайловые программы на C++

# Symbol

## Символы в контексте сборки программы

Функции

Методы

Переменные

\* те, что объявлены в глобальной области видимости

! Классы, структуры, неймспейсы не являются символами

# Declaration VS Definition

## Declaration

Объявление говорит нам о то, что данная сущность существует, но будет определена позже и, может быть, в другом файле

Объявить можно:

- 1) Функцию
- 2) Класс
- 3) Метод класса
- 4) Переменную\*

Объявлений может быть сколько угодно

## Definition

Определение показывает внутреннее устройства сущности (класса, функции, метода)

Определить можно:

- 1) Функцию
- 2) Класс
- 3) Метод класса
- 4) Переменную\*

- Для символов: определение может быть только одно во всей программе\*

- Для классов: сколько угодно в пределах одного файла `cpp`

One definition rule

# Когда необходимо объявление/определение

## Declaration

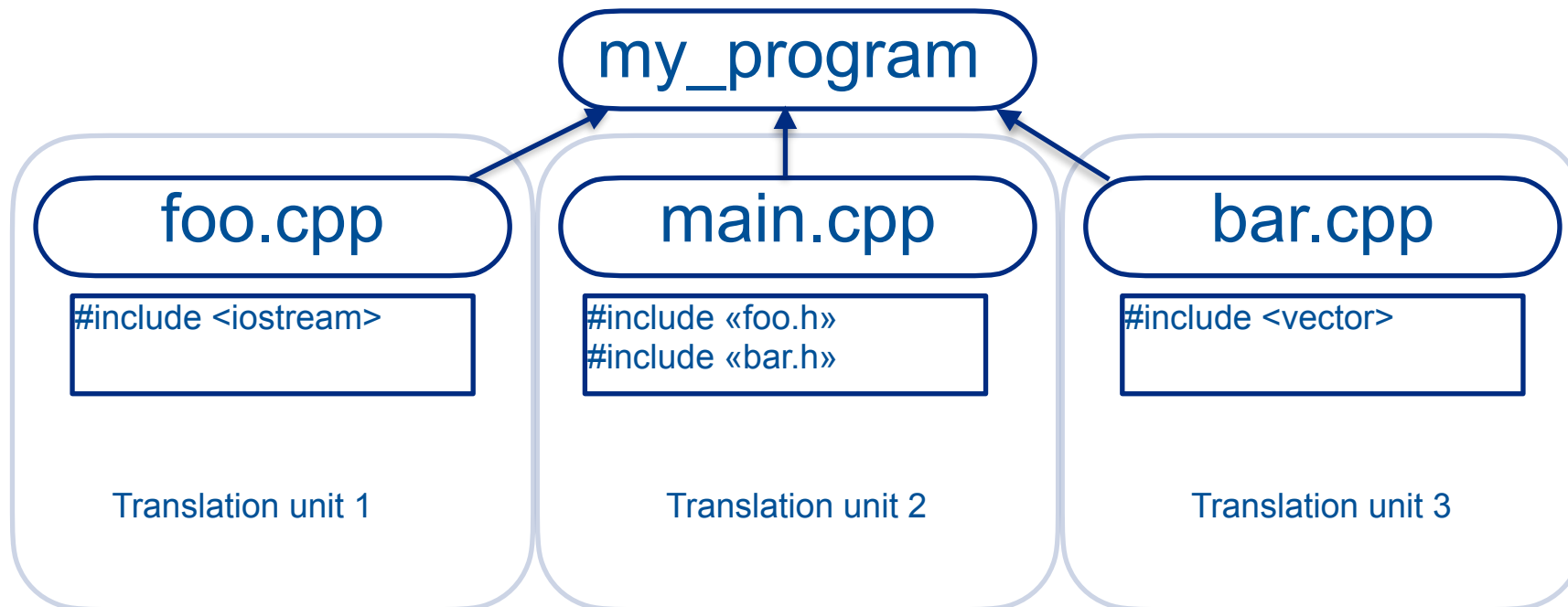
- 1) Необходимо для вызова функций(или методов класса), которые ещё не определены
- 2) Необходимо для класса, который используется в сигнатуре функции, но ещё не определён

## Definition

- 1) Необходимо для объявления переменных, которые являются экземплярами класса



# Из чего состоит программа



## Стадии сборки



Стадия	Действия	Ограниченность действия	Результат
Препроцессинг	Обработка директив препроцессора (всё, что начинается с #)	Одна единица трансляции	Временный файл с раскрытыми макросами и вставленными инклюдами
Компиляция	1) Перевод в ассемблерный код 2) Ассемблирование	Одна единица трансляции	Объектные файлы *.o
Линковка	Детерминирование вызовов функций	Вся программа	Готовая к исполнению программа/библиотека

# Linkage

## Linkage

### External

Видимость функции/  
переменной ничем не  
ограничена

### Internal

Видимость функции/  
переменной ограничена  
одной единицей  
трансляции