

Что будет на уроке

- 1. Познакомимся с понятием директивы препроцессора
- 2. Изучим дополнительные возможности препроцессора C
- 3. Научимся писать функциональные макросы и осуществлять условную компиляцию



Препроцессинг - это процесс анализа и обработки исходного файла программы перед его компиляцией.

Препроцессор осуществляет замену символических выражений значениями согласно директив.

Директива препроцессора - это специальная синтаксическая конструкция, направляющая действия препроцессора и не существующая на следующих шагах компиляции



#define

- Определяет константы;
- Константы могут иметь, а могут не иметь значений;
- Значения констант заменяются в тексте программы;
- Константой может быть вызов функции;
- Обычно константы описываются на одной строке;
- Константы не заменяются внутри строковых литералов;



#define

- Определяет макроподстановки;
- Макросы не являются функциями;
- Макросы подставляются в текст "как есть";
- Нельзя писать имя макроса и аргументы через пробел;
- Всегда пишите достаточное количество скобок;
- У макросов есть специальные операции # и ##;



#define

- Может принимать не строгое число аргументов
- Аргумент переменной длины может быть только последним
- Аргументы макросов не типизированы
- С такими аргументами нужно быть ещё внимательнее



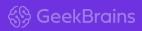
#undef

Отменяет определение макроса или константы, объявленных директивой #define



#error - чаще всего используется для отладки, и осуществляет вывод переданной строки в stderr. В С++ такая директива почти полностью вытеснена механизмом исключений

#pragma - вспомогательная директива препроцессора, нужна для оставления специфичных препроцессорных комментариев. Если препроцессор не распознал значение этой директивы, она просто игнорируется.



Определённые заранее константы и ОС-зависимые константы

- 1. __DATE__ Дата компиляции;
- 2. __FILE__ Имя файла;
- 3. __LINE__ Номер строки;
- 4. __STDC_VERSION__ Версия С;
- 5. __TIME__ Время компиляции;
- 6. _WIN64, _WIN32 индикатор Windows
- 7. LINUX, __linux__ индикатор Linux
- 8. __APPLE__ устройство Apple
- 9. __cplusplus индикатор C++ в Visual Studio



Условная компиляция

Поддержка условий и ветвления для препроцессора Может использоваться в том числе для отладки Позволяет поддерживать кроссплатформенность Позволяет запрещать архитектуры и устаревшие компиляторы

