Inline-функции

inline-функция — это такая функция, чье тело подставляется в каждую точ­ку вызова, вместо того, чтобы генерировать код вызова. Это подобно использованию параметризованных макросов в С. Имеются два способа создания inline-функции. Первый заключа­ется в использовании модификатора inline. Например, для создания inline-функции f, возвращаю­щей значение типа int и не имеющей параметров, достаточно объявить ее следующим образом:

inline int f()  
{

// ..

}

Общая форма объявления inline-функции следующая:

inline объявление\_функции

Модификатор inline предшествует всем частям объявления функции.

Причина использования inline-функции заключается в их эффективности. Всякий раз, когда вы­зывается функция, необходимо выполнить серию инструкций для формирования вызова функции, вставки аргументов в стек и возврата значения из функции. В некоторых случаях для этого прихо­дится использовать много тактов центрального процессора. При использовании inline-функции нет необходимости в таких дополнительных действиях и скорость выполнения программы возрастает.

Когда inline-функция не может быть встроена?

Функция не может быть встроена если:

1. Это рекурсивная функция. Глубина рекурсии в момент компиляции может быть неизвестна, поэтому непонятно сколько раз вставлять тело функции. Даже если глубина известна, встраивание рекурсивной функции приведет к многократному увеличению объема кода.

Некоторые компиляторы(например Visual C++) имеют опции включающие встраивание рекурсивных функций на некоторую фиксированную глубину.

1. Вызов функции по указателю. Если где-то в программе от функции берется указатель, то компилятор создаст тело функции для вызова.  
   При этом если где-то в другом месте функция вызывается непосредственно, то в этом месте она может быть встроена.
2. Виртуальная функция. Если виртуальная функция вызывается через указатель или ссылку на класс, то она вызывается через указатель на функцию в таблице виртуальных функций, поэтому для нее так же должно быть создано тело для обычного вызова.  
   Если виртуальная функция вызывается непосредственно через экземпляр класса, то она может быть встроена.