[Link](https://ru.stackoverflow.com/questions/1330474/%D0%9E%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5-%D0%BC%D0%B0%D1%88%D0%B8%D0%BD%D0%BD%D0%BE%D0%B3%D0%BE-%D1%8D%D0%BF%D1%81%D0%B8%D0%BB%D0%BE%D0%BD-%D0%B4%D0%BB%D1%8F-%D1%82%D0%B8%D0%BF%D0%BE%D0%B2-double-%D0%B8-float-%D0%B2-%D0%A1)   
Не совсем ответ, а скорее заметка по вопросу точности.

Объявляя вещественные переменные как float (32 бит) или double (64 бит), многие почему-то забывают (или не все знают), что процессор (сопроцессор) всегда преобразует эти переменные в 80-битный тип long double - это происходит автоматически при их загрузке в 80-битные регистры стека сопроцессора (при выгрузке значений в память, происходит обратное преобразование). И все вычисления производятся над 80-битными вещественными значениями. Реальная же точность операций задаётся определёнными битами управляющего слова в регистре CWR. По умолчанию эта точность соответствует типу double. А значит, даже если исходные значения хранятся во float, процессор всё равно обрабатывает их в регистрах с точностью double. По-этому, чтобы по-настоящему ограничить точность вычислений в 32 бита, необходимо соответствующим образом изменить значение регистра CWR, либо в ассемблере (с помощью инструкций fstcw, fldcw), либо вызвав библиотечную функцию \_control87().

[FLT\_EVAL\_METHOD 0/2](https://stackoverflow.com/questions/16063820/floating-point-arithmetic-and-machine-epsilon)

Это средство для компилятора/платформы сообщить программисту, как будут оцениваться выражения с плавающей запятой. Вы не должны изменять его значение, и это не должно ни на что влиять.