

Программирование арифметических операций

В том случае, когда значительная часть времени работы программы отводится арифметическим вычислениям, немалые резервы повышения скорости работы программы таются в правильном программировании арифметических (и логических) выражений. Важно, что различные арифметические операции значительно различаются по быстродействию. В большинстве архитектур, самыми быстрыми являются операции сложения и вычитания. Более медленным является умножение, затем идёт деление. Например, вычисление значения выражения $\frac{x}{a}$, где a — константа, для аргументов с плавающей точкой производится быстрее в виде $x \cdot b$, где $b = \frac{1}{a}$ — константа, вычисляемая на этапе компиляции программы (фактически медленная операция деления заменяется быстрой операцией умножения). Для целочисленного аргумента x вычисление выражения $2x$ быстрее произвести в виде $x + x$ (операция умножения заменяется операцией сложения) или с использованием операции сдвига влево (что обеспечивает выигрыш не на всех процессорах).

Быстродействие также зависит и от типа операндов. Например, в языке Turbo Pascal, ввиду особенностей реализации целочисленной арифметики, операция сложения оказывается наиболее медленной для операндов типа `Byte` и `ShortInt`: несмотря на то, что переменные занимают один байт, арифметические операции для них двухбайтовые и при выполнении операций над этими типами производится обнуление старшего байта регистров и операнд копируется из памяти в младший байт регистра. Это и приводит к дополнительным затратам времени.

Программируя арифметические выражения, следует выбирать такую форму их записи, чтобы количество «медленных» операций было сведено к минимуму. Рассмотрим такой пример. Пусть необходимо вычислить многочлен 4-й степени:

$$ax^4 + bx^3 + cx^2 + dx + e$$

При условии, что вычисление степени производится перемножением основания определенное число раз, нетрудно найти, что в этом выражении содержится 10 умножений («медленных» операций) и 4 сложения («быстрых» операций). Это же самое выражение можно записать в виде:

$$(((ax + b)x + c)x + d)x + e$$

Такая форма записи называется схемой Горнера. В этом выражении 4 умножения и 4 сложения. Общее количество операций сократилось почти в два раза, соответственно уменьшится и время вычисления многочлена.