

частоты символов программного кода, которое напомнило ему результаты из опубликованной в 1949 г. работы Г. Зипфа, исследовавшего тексты на естественном языке. На основании этой оказавшейся содержательной аналогии Хэлстед предложил новую статистическую оценку качества программы — логарифмическую меру, которая связывает число активно используемых базовых символов программного кода с «фактором длины» программы. По его мнению, такой фактор длины (length estimator) может объективно характеризовать качество программного кода на всех фазах преобразования текста программы: от языка высокого уровня до ассемблера.

В 1982 г. была опубликована статья Н. Безера «Теоретические и экспериментальные основы программирования»*, в которой автор (кроме обзора избранных работ по этой теме, выполненных в «ИБМ» и других крупнейших фирмах) приводит собственные экспериментальные результаты, полученные обработкой большого объема программных текстов, доказывающие статистическую связь фактора длины с практически интересными характеристиками качества программ, в том числе и ожидаемой вероятностью ошибок [40].

Итак, пока не известно новых критериев качества программ, которые приближались бы по простоте их интуитивного понимания и универсальности применения (а следовательно, и популярности) к «реликтовым» критериям оценки, основанным на прямых измерениях используемых машинных ресурсов: какую вычислительную мощность потребляет программа от всей мощности процессора и какой объем памяти она при этом занимает. Что может быть проще, чем два широкоизвестных аналога основных понятий окружающего нас мира: скорость (операций/с); объем (куб памяти, три куба памяти и т. д.). По-видимому, нелегко будет найти иные критерии, близкие по доходчивости и практически неограниченному кругу применения. Но как оценивать качество прикладных программ потребителю программного продукта — конечному пользователю, если сами программисты пока не в состоянии найти разумную альтернативу «реликтовым» критериям? Где новый ориентир?

* Широковещательное название этой работы (тем не менее в содержательном плане безусловно интересной), видимо, представляет собой в некотором роде дань сложившейся в этой отрасли инженерного искусства традиции, по которой любой сколько-нибудь нетривиальный результат, как правило, оказывается «новомодом для создания очередных «основ науки программирования» («foundations of software science»).