

Не все языки программирования созданы, чтобы решать задачи и помогать людям. Иногда их разрабатывают в шутку или в качестве тренажёра для мозга. Такие языки называют эзотерическими. Расскажем про самый первый из них — INTERCAL.

«Структуры данных? Нам не нужны никакие структуры данных!»

INTERCAL придумали в 1972 году студенты Принстонского университета Джеймс Лайон и Дон Вудс — он задумывался как пародия на существующие «серьёзные» языки.

INTERCAL переводится как «компилируемый язык с непроизносимым акронимом», и в этом вся его суть. Так, базовые операции, знакомые программистам по известным языкам, здесь намеренно написаны с помощью избыточно сложного синтаксиса.

Ничего не понятно? Да никому ничего не понятно! Разработчики на [сайте](#) честно предупреждают, что ничего, «кроме технарских шуток и неумолимого погружения в высасывающую мозг одержимость», при изучении INTERCAL вас не ждёт.

При этом в INTERCAL можно проводить все вычисления, которые доступны в том же C. Правда, очень медленно — на расчёт простых чисел от 1 до 65 536 INTERCAL потребуется 17 часов.

«INTERCAL – простой и полезный язык, особенно для новичка. Он подтолкнёт на поиск другой работы»

Это цитата из мануала, составленного авторами языка. В 1970-е было нормой прикладывать толстые брошюры к программам — они помогали разобраться с продуктом и погрузиться в контекст. У INTERCAL тоже есть такая брошюра — в ней собраны шутки и множество отсылок к «Алисе в Стране чудес». Например, служебный символ “ здесь называется «кроличьи ушки», а каждая глава снабжена цитатой Льюиса Кэрролла.

Ещё одна интересная особенность INTERCAL — коды ошибок и их формулировки. Они здесь не помогают, а скорее путают:

- «Не помню, что хотел сказать» — память при вводе-выводе оказалась переполнена.
- «Эта метка, кажется, тебе очень нравится» — метка использована несколько раз.
- «Программа не выполнена ради сохранения рассудка» — команды многопоточности использованы без компилятора.

Есть и уникальные ошибки — их бережно сохраняют и переносят в новые реализации языка:

- «Программист слишком неучтив» и «Программист слишком учтив» — команда PLEASE использована слишком редко или слишком часто.
- «Случайный баг компилятора» — ошибка возникает случайно со стороны кода, исправить её можно только повторной компиляцией.
- «Как я это должен сделать?» — абсурдная ошибка, возникающая при запуске кода, которого нет.

Странности INTERCAL не заканчивались и на операторах. Например, здесь используют COME FROM («Перейди из»), FORGET («Забудь») и парадоксальный PLEASE ABSTAIN FROM CALCULATING («Пожалуйста, воздержись от расчётов»).

Язык-шутка, придуманный студентами в 1970-х, стал идейной основой для всех последующих эзотерических языков, которые становились всё изощрённее и сложнее.