

## **§6. Тест программы**

### **6.1. Тест и тестирование**

Как бы ни была тщательно отлажена программа, решающим этапом, устанавливающим ее пригодность для работы, является контроль программы по результатам ее выполнения на системе тестов.

Программу условно можно считать правильной, если её запуск для выбранной системы тестовых исходных данных во всех случаях дает правильные результаты.

Но, как справедливо указывал известный теоретик программирования Э. Дейкстра, тестирование может показать лишь наличие ошибок, но не их отсутствие. Нередки случаи, когда новые входные данные вызывают «отказ» или получение неверных результатов работы программы, которая считалась полностью отлаженной.

Для реализации метода тестов должны быть изготовлены или заранее известны эталонные результаты. Вычислять эталонные результаты нужно обязательно **до**, а не **после** получения машинных результатов. В противном случае имеется опасность невольной подгонки вычисляемых значений под желаемые, полученные ранее на машине.

### **6.2. Тестовые данные**

Тестовые данные должны обеспечить проверку всех возможных условий возникновения ошибок:

- должна быть испытана каждая ветвь алгоритма;
- очередной тестовый прогон должен контролировать нечто такое, что еще не было проверено на предыдущих прогонах;
- первый тест должен быть максимально прост, чтобы проверить, работает ли программа вообще;
- арифметические операции в тестах должны предельно упрощаться для уменьшения объема вычислений;
- количества элементов последовательностей, точность для итерационных вычислений, количество проходов цикла в тестовых примерах должны задаваться из соображений сокращения объема вычислений;
- минимизация вычислений не должна снижать надежности контроля;
- тестирование должно быть целенаправленным и систематизированным, так как случайный выбор исходных данных привел бы к трудностям в определении ручным способом ожидаемых результатов;

кроме того, при случайном выборе тестовых данных могут оказаться непроверенными многие ситуации;

- усложнение тестовых данных должно происходить постепенно.

### **6.3. Этапы процесса тестирования**

Процесс тестирования можно разделить на три этапа.

**1. Проверка в нормальных условиях.** Предполагает тестирование на основе данных, которые характерны для реальных условий функционирования программы.

**2. Проверка в экстремальных условиях.** Тестовые данные включают граничные значения области изменения входных переменных, которые должны восприниматься программой как правильные данные. Типичными примерами таких значений являются очень маленькие или очень большие числа и отсутствие данных. Еще один тип экстремальных условий – это граничные объемы данных, когда массивы состоят из слишком малого или слишком большого числа элементов.

**3. Проверка в исключительных ситуациях.** Проводится с использованием данных, значения которых лежат за пределами допустимой области изменений. Известно, что все программы разрабатываются в расчете на обработку какого-то ограниченного набора данных. Поэтому важно получить ответ на следующие вопросы:

- что произойдет, если программе, не рассчитанной на обработку отрицательных и нулевых значений переменных, в результате какой-либо ошибки придется иметь дело как раз с такими данными?
- как будет вести себя программа, работающая с массивами, если количество их элементов повысит величину, указанную в объявлении массива?
- что произойдет, если числа будут слишком малыми или слишком большими?

Наихудшая ситуация складывается тогда, когда программа воспринимает неверные данные как правильные и выдает неверный, но правдоподобный результат.

Программа должна сама отвергать любые данные, которые она не в состоянии обрабатывать правильно.