Основные методы обеспечения качества функционирования.

Надежность компьютерных систем – это способность системы сохранять работоспособность при воздействии различных внешних и внутренних факторов. При этом не смотря на использование высококачественного оборудования и надежных технологий, невозможно полностью исключить вероятность сбоев.

Существует следующие оперативные методы повышения надежности с использованием различных типов избыточности: **временной**, **информационная** и **программная**.

Временная избыточность заключается в том, что система выполняет одну и ту же задачу несколько раз за определенный промежуток времени, при этом результаты этих операций сравниваются для выявления возможных ошибок.

**Принцип работы временной избыточности**

Этот метод предполагает, что система выполняет операцию несколько раз с некоторыми промежутками, чтобы оценить ее стабильность и точность. Результаты нескольких операции, если они совпадают, считается правильными. В случае, если выводы отличаются, система может заново выполнить операцию или предпринять меры для исправления ошибки.

**Применение временной избыточности**

**Контроль ошибок при вычислениях**

**Контроль ошибок при вычислениях:** Например, в системах с критической точностью, таких как астрономические вычисления или операции в научных вычислениях, используется многократное выполнение одной и той же операции.

**Отложенная проверка результатов:** Это может быть полезно в реальном времени, например, в системах мониторах, где для получения точных данных могут быть выполнены повторные измерения с интервалами.

**Преимущества:** Временная избыточность помогает минимизировать вероятность ошибок, связанных с неожиданными сбоями или помехами в процессе вычислений.

**Ограничения:** Этот метод требует дополнительных вычислительных ресурсов и времени, что может привести к увеличению затрат на обработку данных и замедлению реакции системы.

**Информационная избыточность**

Информационная избыточность предполагает добавление дополнительной информации в систему для улучшения ее надежности. Этот метод помогает минимизировать ошибки, связанные с потерей или повреждением данных, и может использоваться в случае передачи данных по ненадежным каналам или в процессе хранения.

**Принцип работы информационной избыточности**

Информационная избыточность достигается за счет использования дополнительных данных, которые могут быть использованы для восстановления поврежденных или утерянных данных.

Эти данные могут быть добавлены на разных уровнях:

* **Кодирование данных:** например, использование избыточной битов в кодах Хемминга для исправления ошибок в передаваемых данных
* **Резервное копирование:** создание нескольких копий данных для предотвращения их потери в случае сбоя.

**Использование паритетных битов:** в некоторых вычислительных системах для обнаружения и исправления ошибок используются дополнительные биты, которые позволяют восстановить информацию, если она была повреждена.

**Применение информационной избыточности**

**Массивы RAID:** это метод хранения данных, который использует избыточность для повышения надежности. Например, RAID 1(зеркалирование) создает копии данных на разных устройствах хранения, что позволяет восстановить информацию в случае сбоя одного из носителей.

**Системы с контрольными суммами:** в ПО для обеспечения целостности данных часто используется контрольная сумма, которая добавляется к данным для последующей проверки их целостности.

**Преимущества и ограничения**

**Преимущества:** информационная избыточность значительно увеличивает устойчивость системы к сбоям и потерям данных

**Ограничения:** она требует дополнительных ресурсов для хранения и передачи данных, что может повлиять на производительность системы. Также существует вероятность, что избыточные данные могут быть также повреждены в случае серьезного сбоя.

**Программная избыточность**

Программная избыточность заключается в дублировании программных компонентов, что позволяет системе продолжать функционировать при сбоях или ошибках в одном из компонентов.

**Принцип работы программной избыточности**

Программная избыточность достигает за счет использования нескольких экземпляров программных компонентов, которые выполняют одни и те же задачи. В случае сбоя одного компонента другой компонент продолжает выполнять задачу, обеспечивая бесперебойную работу системы.