

НАДЕЖНОСТЬ ИНФОРМАЦИОННЫХ СИСТЕМ

направление 09.03.02

«Информационные системы и
технологии»

Надежность программного обеспечения

Лекция 4

Надежность программного обеспечения

Надежность программного обеспечения (ПО) – свойство программы выполнять заданные функции, сохранять свои характеристики в установленных пределах при определенных условиях эксплуатации.

Безотказность программы или программного обеспечения есть его (ее) свойство сохранять работоспособность при использовании в процессе обработки информации на компьютере.

Восстанавливаемость определяется затратами времени и труда на устранение отказа из-за проявившейся ошибки в программе и его

Вопрос 1

Сравнительная характеристика аппаратных и программных отказов

Сравнительная характеристика аппаратных и программных отказов

Общее между программными и аппаратными отказами:

- а) невыполнение объектом заданных функций;
- б) времена до отказов и времена устранения отказов носят случайный характер;
- в) методы обработки статистических данных об отказах одинаковы, а потому статистические оценки показателей надёжности аппаратной и программной, полученные по результатам испытаний и эксплуатации, могут быть одинаковыми по своему названию: средняя наработка объекта на программный отказ, интенсивность программных отказов объекта и т.д.

Возможны и объединённые (комплексные) оценки: средняя наработка объекта на программный и аппаратный отказ и т.п.

Сравнительная характеристика аппаратных и программных отказов

Программные отказы существенно отличаются от отказов аппаратных:

а) отказ аппаратный зависит либо от времени, либо от объёма выполненной работы, а отказ программный – от той функции, которую выполняет изделие под управлением программы;

б) обнаружение и устранение аппаратного отказа не означает, что такой же отказ не повторится при дальнейшей работе изделия, а обнаружение и устранение отказа программного означает, что такой отказ в дальнейшем не повторится;

Сравнительная характеристика аппаратных и программных отказов

Программные отказы существенно отличаются от отказов аппаратных:

в) программный отказ, обнаруживаемый при автономной проверке программы, может переходить в разряд недействующих, если состояние аппаратуры делает её нечувствительной к данному виду программного отказа.

г) прогнозировать возникновение аппаратных отказов сравнительно легко, а прогнозировать возникновение отдельных программных отказов трудно, а часто и невозможно;

Сравнительная характеристика аппаратных и программных отказов

Программные отказы существенно отличаются от отказов аппаратных:

д) аппаратные отказы целесообразно подразделять на внезапные и постепенные, т.е. отказы, различные по своей физической природе, законам распределения времени до отказа, методам борьбы за снижение их вероятности.

Программные отказы нет смысла делить на внезапные и постепенные. Они возникают, как только программа переходит на такой участок, который содержит "ошибку". В то же время они по природе своей не совпадают с внезапными аппаратными отказами.

Вопрос 2

Основные причины отказов ПО

Основные причины отказов ПО

Основные причины непосредственно вызывающие нарушение нормального функционирования программы:

1. Ошибки, скрытые в самой программе.
2. Искажения входной информации, подлежащей обработке.
3. Неверные действия пользователя.
4. Неисправность аппаратуры установки, на которой реализуется вычислительный процесс.

Основные причины отказов ПО

1. *Скрытые ошибки* программы являются главным фактором нарушения нормальных условий его функционирования.

Основные ошибки в программе:

- Ошибки вычислений
- Логические ошибки
- Ошибки ввода-вывода
- Ошибки манипулирования данными
- Ошибки совместимости
- Ошибки сопряжений

Основные причины отказов ПО

2. *Искажения информации*, подлежащей обработке, вызывает нарушение функционирования ПО, когда входные данные не попадают в область допустимых значений переменных программы.

Причины искажения вводимой информации:

- искажения данных на первичных носителях информации;
- сбои и отказы в аппаратуре ввода данных с первичных носителей информации;
- шумы и сбои в каналах связи при передаче сообщений по линиям связи и т.д.

Основные причины отказов ПО

3. *Неверные действия* пользователя, приводящие к отказу в процессе функционирования ПО связаны, прежде всего:

- с неправильной интерпретацией сообщений,
- неправильными действиями пользователя в процессе диалога с компьютером и т.д.

Основные причины отказов ПО

4. *Неисправность аппаратуры* – неисправности, возникающие при работе аппаратуры, используемой для реализации вычислительного процесса, оказывают влияние на характеристику надежности ПО.

Появление отказа или сбоя в работе аппаратуры приводит к нарушению нормального хода вычислительного процесса и во многих случаях к искажению данных и текстов программ в основной и внешней памяти.

Основные причины отказов ПО

Признаки появления ошибок:

- преждевременное окончание выполнения программы;
- недопустимое увеличение времени некоторой последовательности команд одной из программ;
- полная потеря или значительное искажение накопленных данных, необходимых для успешного выполнения решаемых задач;

Основные причины отказов ПО

Признаки появления ошибок:

- нарушение последовательности вызова отдельных программ, в результате чего происходит пропуск необходимых программ;
- искажение отдельных элементов данных (входных, выходных, промежуточных) в результате обработки искаженной исходной информации.

Вопрос 3

Основные показатели и модели надежности ПО

Основные показатели надежности ПО

1. *Вероятность безотказной работы $P(t_3)$* – это вероятность того, что в пределах заданной наработки отказ системы не возникает.

Основные показатели надежности ПО

2. *Вероятность отказа* – вероятность того, что в пределах заданной наработки отказ системы возникает.

Это показатель, обратный предыдущему.

$$Q(t_3) = 1 - P(t_3)$$

где t_3 – заданная наработка, ч.;

$Q(t_3)$ – вероятность отказа.

Основные показатели надежности ПО

3. *Интенсивность отказов системы* – это условная плотность вероятности возникновения отказа ПО в определенный момент времени при условии, что до этого времени отказ не возник.

$$\lambda(t) = \frac{f(t)}{P(t)}$$

где $f(t)$ – плотность вероятности отказа в момент времени t .

Основные показатели надежности ПО

4. Средняя наработка на отказ T_i – математическое ожидание времени работы ПО до очередного отказа:

$$T_i = \int_0^t t \cdot f(t) dt$$

Иначе среднюю наработку на отказ T_i можно представить:

$$T_i = \frac{t_1 + t_2 + t_3 + \dots + t_n}{n} = \left(\frac{i}{n}\right) \sum_{i=1}^n t_i$$

где t – время работы ПО между отказами, с.

n – количество отказов.

Основные показатели надежности ПО

5. Среднее время восстановления T_B – математическое ожидание времени восстановления;

- времени, затраченного на обнаружение и локализацию отказа – t_o ;
- времени устранения отказа – t_y ;
- времени пропускной проверки работоспособности – $t_{пр}$:

$$t_{Bi} = t_o + t_y + t_{пр},$$

где t_{Bi} – время восстановления после i -го отказа

$$nT = \frac{i}{nt} \Rightarrow i = 1$$

где n - количество отказов.

Основные показатели надежности ПО

6. Коэффициент готовности K_{Γ} – вероятность того, что ПО ожидается в работоспособном состоянии в произвольный момент времени его использования по назначению:

$$K_{\Gamma} = \frac{T_p}{T_p + T_B}$$

где T_p – суммарное время нахождения объекта в работоспособном состоянии;

T_B – суммарное время восстановления объекта.

Основные показатели надежности ПО

7. Средняя тяжесть ошибок (СТО):

$$B = \frac{1}{Q \sum_{i=1}^m (b_i \cdot p_i \cdot z_i)}$$

где Q – вероятность сбоя ПО;

$\sum_{i=1}^m$ – оператор суммирования по переменной i ;

b_i – функция принадлежности тяжести последствий ошибки, возникшей при i -ом наборе входных данных, к максимально тяжелым последствиям;

p_i – вероятность ввода i -го набора входных данных при эксплуатации ПО;

z_i – дихотомическая переменная, равная 1, если при i -ом наборе входных данных был зафиксирован сбой, и 0 в противном случае;

m – общее число наборов входных данных.

Модели надежности ПО

Аналитические модели дают возможность рассчитывать количественные показатели надежности, основываясь на данных о поведении программы в процессе тестирования.

Эмпирические модели базируются на анализе структурных особенностей программ.

Модели надежности ПО

В *динамических* поведении ПО (появление отказов) рассматривается во времени.

В *статических* моделях появление отказов не связывают со временем, а учитывают только зависимость количества ошибок от числа тестовых прогонов или зависимость количества ошибок от характеристики входных данных.

Динамические модели надежности

Модель Шумана

Модель Шумана может быть использована при определенном образе организованной процедуре тестирования.

Использование модели Шумана предполагает, что тестирование проводится в несколько этапов.

При использовании модели Шумана предполагается, что исходное количество ошибок в программе постоянно, и в процессе тестирования может уменьшаться по мере того, как ошибки выявляются и исправляются.

Динамические модели надежности

Модель Шумана

Предполагается, что до начала тестирования в ПО имеется E_t ошибок. В течении времени тестирования t_I в системе обнаруживается E_c ошибок в расчете на команду в машинном языке.

Таким образом, удельное число ошибок на одну машинную команду, оставшуюся в системе после t_I времени тестирования, равно:

$$E(t) = \frac{E_t}{I_t} E(t)_c$$

где I_t – общее число машинных команд, которое предполагается в рамках этапа тестирования.

Динамические модели надёжности

Модель *La Padula*

По этой модели выполнение последовательности тестов производится в m этапов. Каждый этап заканчивается внесением изменений в ПО.

Возрастающая функция надёжности базируется на числе ошибок, обнаруженных в ходе каждого тестового прогона.

Надёжность ПО в течении i –го этапа:

$$R(t) = R(\infty) - \frac{A}{(i)}$$

где $i = 1, 2, \dots, n$,

A – параметр роста.

Статические модели надежности

Статические модели принципиально отличаются от динамических прежде всего тем, что в них не учитывается время появления ошибок в процессе тестирования и не используется никаких предположений о поведении функции риска.

Эти модели строятся на твердом статическом фундаменте.

Статические модели надежности

Модель Миллса

Соотношение:

$$N = \frac{S \cdot n}{V}$$

дает возможность оценить N – первоначальное количество ошибок в программе.

В данном соотношении, которое называется формулой Миллса:

S – количество искусственно внесенных ошибок,

n – число найденных собственных ошибок,

V – число обнаруженных к моменту оценки искусственных ошибок.

Статические модели надежности

Модель Миллса

Предположим, что в программе имеется K собственных ошибок, и внесем в нее еще S ошибок. В процессе тестирования были обнаружены все S внесенных ошибок и n собственных ошибок.

Тогда по формуле Миллса мы предполагаем, что первоначально в программе было $N = n$ ошибок.

Вероятность возможно рассчитать по следующему соотношению:

$$C = 1, \text{ если } n > K$$
$$C = \frac{S}{S+K+1}, \text{ если } n \leq K$$

Статические модели надежности

Модель Миллса

Когда оценка надежности производится до момента обнаружения всех S рассеянных ошибок, величина C рассчитывается по модифицируемой формуле

$$C = 1, \text{ если } n > K$$

$$C = \frac{\frac{S}{V-1}}{\left[\frac{S+K+1}{V+K} \right]}, \text{ если } n \leq K$$

где числитель и знаменатель формулы при $n \leq K$ являются биномиальными коэффициентами вида

$$\frac{a}{b} = \frac{a!}{b! (a-b)!}$$

Статические модели надежности

Модель Миллса

Например, если утверждается, что в программе нет ошибок, а к моменту оценки надежности обнаружено 5 из 10 рассеянных ошибок и не обнаружено ни одной собственной ошибки, то вероятность того, что в программе действительно нет ошибок, будет равна:

$$C = \frac{\frac{10}{4}}{\frac{11}{5}} = \frac{10! \cdot 5! \cdot 6!}{4! \cdot 6! \cdot 11!} = 0,45$$

Статические модели надежности

Модель Липова

Собственные и искусственные ошибки имеют равную вероятность быть найденными, то вероятность обнаружения n собственных и V внесенных ошибок равна:

$$Q(n, V) = \frac{m}{n + V} \cdot q^{n+V} \cdot (1 - q)^{m - n - \frac{V \cdot \frac{N}{n} \cdot \frac{S}{V}}{\left(\frac{N+S}{n+V}\right)}}$$

где m – количество тестов, используемых при тестировании;

q – вероятность обнаружения ошибки в каждом из m тестов, рассчитывается по формуле:

$$q = \frac{n + V}{n}$$

S – общее количество искусственно внесенных ошибок;

N – количество собственных ошибок, имеющих в ПО до начала тестирования.

Статистические модели надежности

Модель Липова

Для использования модели Липова должны выполняться следующие условия:

$$N \geq n \geq 0$$

$$S \geq V \geq 0$$

$$m \geq n + V \geq 0$$

Модель Липова дополняет модель Миллса, дав возможность оценить вероятность обнаружения определенного количества ошибок к моменту оценки.