



ТЕСТВАНЕ С КОНТРОЛНИ СПИСЪЦИ И КЛАСОВЕ НА ЕКВИВАЛЕНТНОСТ

доц. д-р Десислава Петрова-Антонова

Съдържание

- ❖ Тестване с контролни списъци
- ❖ Тестване с класове на еквивалентност
- ❖ Тестване, базирано на употреба с оперативни профили
- ❖ Разработване на оперативен профил: процедури и примери

Тестов контролен списък

❖ “ad hoc” тестване

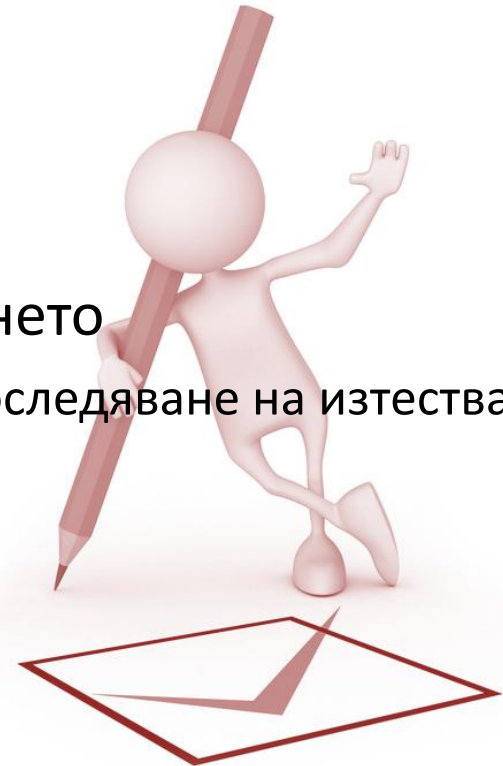
- Стартиране на софтуера, наблюдение на поведението и идентифициране на проблеми

❖ Проследяване на “ad hoc” тестването и създаване на формални “to do” списъци

- Физически списъци
- Онлайн списъци
- Мисловен списък

❖ Систематизиране на “ad hoc” тестването

- Създаване на контролни списъци за проследяване на изтестваните елементи



Тестване с контролен списък

❖ Примерни контролни списъци

- Елементи на спецификацията
- Характеристики на системата, които ще бъдат реализирани
- Стандарти за програмиране, които се очаква да бъдат следвани
- Изрази от програмен код
- Компоненти, които ще бъдат интегрирани в системата
- Оперативен профил

❖ Условие за спиране на тестването

- Изчерпване на елементите в списъка
 - ✓ Изпълнение на тестов сценарий и приключване на дейностите по отстраняване на дефекти

❖ Често използвани контролни списъци

- Функционални контролни списъци с различно ниво на абстракция и гранулярност
- Контролни списъци със системни елементи с различно ниво на гранулярност
- Контролни списъци със структури или характеристики, които се споделят от различни елементи (модули, ресурси и др.)
- Контролни списъци с определени свойства

Примерен функционален контролен списък

- ☒ Abnormal termination
- ☒ Backup and restore
- ☒ Communication
- ☒ Co-existence
- ☒ File I/O
- ☒ Gateway
- ☒ Index management
- ☒ Installation
- ☒ Logging and recovery
- ☒ Locking
- ☒ Migration
- ☒ Stress



Типове контролни списъци

❖ Базови списъци

- Включват набор от елементи, които се тестват

❖ Йерархични списъци

- Всеки елемент от контролен списък на по-горно ниво съдържа контролен списък от по-ниско ниво като нивото на детайност се определя спрямо критерий

❖ Комбинирани списъци

- Създаване на многомерни списъци, при които всеки контролен списък се обхожда за всички елементи от останалите контролни списъци

❖ Смесени списъци

- Съчетават йерархичните и комбинираните списъци

Типове контролни списъци: пример

Базов списък	
☒	Abnormal termination
☒	Backup and restore
☒	Communication
☒	Co-existence
☒	File I/O
☒	Gateway
☒	Index management
☒	Installation
☒	Logging and recovery
☒	Locking
☒	Migration
☒	Stress

Component	Standards Items			
	S_1	S_2	...	S_n
C_1	☒	☒		☒
C_2	☒	☒		☒
...				
C_n	☒	☒		☒



Проблеми и ограничения при контролните списъци

- ❖ Структурната и функционалната сложност на съвременните системи ограничава използването на контролните списъци
 - Трудности при създаването на тестови сценарии от контролния списък
 - Съхраняване на информация за средата извън контролния списък
 - Трудности при диагностика на проблема
- ❖ Проблеми и ограничения
 - **Трудности при покриване на всички функционални или структурни компоненти** от различни гледни точки и при различни нива на гранулярност
 - Опастност от **припокриване на елементи в различните контролни списъци**
 - **Трудности при описание на сложни взаимодействия** между различните системни компоненти или между главните системни функции
- ❖ Алтернативи
 - Тестване с покритие на класове (проблеми 1 и 2)
 - ✓ По-прецизно дефиниране на тестови сценарии
 - Тестване с машина на крайните (проблем 3)

Инициализация на тестовите сценарии: мотивация

- ❖ Инициализиране на тестови сценарии при елементарни входно-изходни взаимовръзки
- ❖ Пример 1: Взимане на решение въз основа на стойностите на булеви променливи
 - Броят на входните комбинации е 2^n за n на брой променливи
- ❖ Пример 2: Изчисляване на квадратно уравнение

$$ax^2 + bx + c = 0, \quad r = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

- При представянето на променливите a , b и c с 32-битово число с плаваща запетая броят на входните комбинации е $2^{32} \times 2^{32} \times 2^{32} = 2^{96}$
- Разделяне на входния домейн и дефиниране на тестови сценарии посредством използване на зависимостта между променливите r и d

ТС	Условие $d = b^2 - 4ac$	Вход		
		a	b	c
1	$d > 0$	1	2	-1
2	$d = 0$	1	2	1
3	$d < 0$	1	2	3

Тестване с разделяне на класове

Дефиниране на
класове на
еквивалентност

Избор на един
тестов сценарии
за всеки клас на
еквивалентност

Постигане на
пълно покритие
на класовете на
еквивалентност

Разделяне на класове: концепции и дефиниции

❖ Формално представяне на клас на еквивалентност

- Множество S с подмножества $G_1, G_2, \dots, G_n$, със следните свойства

- ✓ Подмножествата са взаимно изключващи се, т.е. не споделят общи елементи

$$\forall i, j, i \neq j \Rightarrow G_i \cap G_j = \emptyset$$

- ✓ Помножествата са взаимно изчерпателни, т.е. покриват всички елементи на оригиналното множество

$$\bigcup_{i=1}^n G_i = S$$

- ✓ Елементите на всяко подмножество са еднакви от гледна точка на специфично свойство, използвано за дефинирането му

❖ Релации между елементите в класовете на еквивалентност

- Симетрична релация: запазва се, ако редът на елементите се промени (" $=$ ")

- ✓ $R(a,b) \Rightarrow R(b,a)$

- Транзитивна релация: запазва се в релационна верига (" $>$ ")

- ✓ $R(a,b) \wedge R(b,c) \Rightarrow R(a,c)$

- Рефлексна релация: запазва се върху всеки елемент сам по себе си (" $=$ ")

- ✓ $R(a,a)$

Типове разделяне на класове

- ❖ Разделяне, базирано на софтуерни елементи, като например външни функции или системни компоненти
 - Необходимо е да се гарантира, че подмножествата покриват напълно елементите на оригиналното множество и не се припокриват
 - Пример: Разделяне на компоненти по под-системи ($x \in S$)
- ❖ Разделяне, базирано на определени свойства, релации и логически условия
 - Директно използване на логически предикати посредством логически променливи и свързани с логически оператори AND, OR или NOT
 - Сравнение на числови стойности посредством оператори за сравнение като “<”, “>”, “=”, “≠”, “≥” и “≤”, например $S_1 \equiv \{x : x < 0\}$ и $S_2 \equiv \{x : x \geq 0\}$
- ❖ Комбинации от представените типове
 - Пример: поддомейн на положителните цели числа по-малки от 21 се специфицира като $(x \in I) \wedge (x > 0) \wedge (x < 21)$

Дърво и таблици за взимане на решения

- ❖ Тестването, базирано на решения като част тестването с разделяне на класове
 - Представяне на изпълнението на програмата като последователност от взимане на решения, започваща от начално решение и приключваща с крайно решение за изход
- ❖ Генериране на тестови сценарии от дърво за взимане на решение
 - Тълкуване на дървото като йерархичен контролен списък
 - ✓ Всеки път в дървото се тълкува като под-домейн
 - ✓ Всяка точка за взимане на решение се тълкува като елемент в под-списък
- ❖ Постигане на тестово покритие за всички решения
 - Пример: Покритие при решения, базирани на логически променливи
 - ✓ $P \wedge Q$ (TT), $P = \text{True}$ and $Q = \text{True}$
 - ✓ $P \wedge \neg Q$ (TF), $P = \text{True}$ and $Q = \text{False}$
 - ✓ $\neg P \wedge Q$ (FT), $P = \text{False}$ and $Q = \text{True}$
 - ✓ $\neg P \wedge \neg Q$ (FF), $P = \text{False}$ and $Q = \text{False}$
 - Пример: Покритие при решения, базирани на целочислени променливи ($x > 0$) $\wedge$ ($x < 21$)
 - ✓ $\{1, 10, 100, \dots\}$
 - ✓ $\{1, 10\}$
 - ✓ $x = 10$

Статистическо тестване, базирано на употреба

❖ Дефиниция за **надеждност**

- Вероятност за отсъствие на повреди в операциите на системата за определен период от време или специфично входно множество

❖ Тестване на продукта по начин, наподобяващ използването му от крайните потребители

- Събиране на информация относно сценариите на използване, шаблоните и честотата на използване от крайните потребители
- Анализирание на събраната информация и организирането ѝ в модел, наречен **оперативен профил**
- Извършване на тестване в съответствие с оперативния профил
- Използване на тестовите резултати за оценка на надеждността на системата и осигуряване на обратна връзка

❖ Тестването, базирано на употреба, може да започне на по-късните тестови фази

- Създаване на оперативен профил на ниво продуктово планиране или събиране на изискванията към системата
- Извършване на тестване спрямо оперативния профил след реализацията на повечето софтуерни компоненти

Оперативен профил на Муса: Основна идея

- ❖ **Определение за оперативен профил**
 - Списък, съдържащ множество от операции и техните вероятности да се използват
 - Количествена характеристика за начина, по който системата се използва
- ❖ Всяка операция се асоциира с множество тестови сценарии или тестови серии
- ❖ Всяка операция представя отделен клас на еквивалентност
- ❖ Създаването на оперативни профили с малка гранулярност е непрактично



File type	Hits	% of total
.gif	438536	57,47
.html	128869	16,89
directory	87067	11,41
.pdf	65876	8,63
.class	10784	1,41
.PS	10055	1,32
.PPT	2737	0,36
.jpg	2510	0,33
.css	2008	0,26
.txt	1597	0,21
.doc	1567	0,21
.ico	1254	0,16
.C	849	0,11

Ключови моменти, свързани с оперативния профил

- ❖ Сортиране на операциите в намаляващ ред на вероятността да бъдат използвани
- ❖ Неравномерно разпределение на вероятностите за използване на операциите
- ❖ Дефиниране на вероятностен праг в оперативния профил
 - Премахване на операциите с много малка вероятност да бъдат използвани
- ❖ Препоръчително представяне на оперативния профил с хистограма

Приложение на оперативния профил

- ❖ Статистическо тестване с оперативен профил
 - Вероятностите за използване на операциите определят минималния брой тестови сценарии, които трябва да бъдат създадени или избрани
 - Броят на тестовите серии за всяка операция е пропорционален на нейната вероятност за използване
- ❖ Прогресивно тестване с оперативен профил
 - Повишаване на вероятностния праг в оперативния профил и избор на най-често използваните операции за тестване в началото на тестовия процес
- ❖ Количествена оценка на надеждността с оперативен профил
 - Данните от статистическото тестване с оперативен профил могат да се анализират с модели за надеждност, при което да се оцени дали продуктът е готов за използване от крайните потребители
- ❖ Подобряване на продуктивността и намаляване на разходите
 - Намаляване на разходите за системно тестване с 56%
- ❖ Ускорено представяне на нов продукт
 - Приоритетно реализиране на най-често използваната функционалност
- ❖ Подобрена комуникация и връзка с потребителите на софтуера
- ❖ Висока възвръщаемост на инвестициите
 - Разработването на оперативен профил за продукт с 1000 реда програмен код, който се изработва от 10 разработчика за 18 месеца, изисква разход за 1 човеко-месец
 - Възвръщаемостта от гледна точка на цена-полза е 1:10

Брой на оперативните профили за даден продукт



Методи за създаване на оперативен профил

- ❖ Действително измерване на използваемостта при потребителите
 - Висока точност на създадения оперативен профил
 - Невъзможност за приложение при нови продукти и нови приложения
 - ✓ Използват се подобни продукти или предходни версии на продукта за различни приложения
 - Наличие на ограничения при използване на корпоративни и лични данни
 - ✓ Извършване на тестове от избрани потребители
 - Висока цена на действителното измерване
- ❖ Проучване сред целевите потребители
 - При комерсиалните продукти и продуктите за масовия пазар е необходимо да се балансира между цената и точността на проучването
 - Точността на проучването се влияе от познанията на потребителите за функционалността на продукта
- ❖ Експертна оценка на използваемостта
 - Ниска цена на създаване за сметка на точността, която може да бъде валидирана на по-късен етап с действително измерване или проучване

Метод на Муса за създаване на оперативен профил

❖ Служители в отдел по планиране и маркетинг

- Осигуряват отчитане на потребителските гледни точки в оперативния профил
- Инициализират събирането на изискванията към продукта, които се отчитат в оперативния профил

❖ Системни инженери

- Отговорни са за дефиниране на потребителските изисквания и спецификацията на продукта
- Осигуряват адекватно представяне на системата чрез оперативен профил

❖ Проектанти на високо ниво

- Разработват общ проект на продукта въз основа на спецификацията
- Подпомагат свързването на външните функции на продукта, очаквани от потребителите, с вътрешните компоненти, модули и подсистеми на продукта, които се тестват

Процедура за разработване: Муса-1

❖ Подходяща за големи системи с хетерогенни потребители

❖ Стъпки на разработване

- Определяне на клиентски профил

- ✓ Дефиниране на цялостно множество от клиентски категории с теглови коефициенти, определен от фактор на използване

- Създаване на потребителски профил

- ✓ Дефиниране на потребителски типове с теглови коефициенти, съответстващи на клиентските групи и относителните степени на използване

- Дефиниране на системни режими и съответстващ профил

- ✓ Идентифициране на множества от операции, които се използват съвместно и определяне на съответстващите им теглови коефициенти

- Създаване на функционален профил

- ✓ Идентифициране на индивидуалните функции от високо ниво и определяне на съответстващите им теглови коефициенти въз основа на анализ на системните режими и променливи на средата

- Създаване на оперативен профил

- ✓ Транслиране на функциите от високо ниво в детайлни операции и определяне на тяхната вероятност за използване

ПРИМЕР: Муса-1 профил

Клиентски профил

Customer Type	Weight
corporation	0,5
government	0,4
education	0,05
other	0,05

Потребителски профил

User type	User profile by customer type				Overall user profile
	Customer type				
	com	gov	edu	etc	
	Weight				
	0,5	0,4	0,05	0,05	
end user	0,8	0,9	0,9	0,7	0,84
dba	0,02	0,02	0,02	0,02	0,02
programmer	0,18	-	-	0,28	0,104
third party	-	0,08	0,08	-	0,036

❖ Клиентски групи

- Формират се от организациите, за които е предназначен продукта
- Тегловите коефициенти се определят в зависимост от броя на организациите във всяка група и тяхното значение за доставчика на продукта

❖ Потребителски групи

- Формират се в рамките на клиентските групи от потенциалните потребители на продукта
- End user: $0,8 \times 0,5$ (com) + $0,9 \times 0,4$ (gov) + $0,9 \times 0,05$ (edu) + $0,7 \times 0,05$ (etc) = 0,84

❖ Системни режими

- Режим на поддръжка с операции за архивиране на базата от данни и диагностика

❖ Функционален профил

- Определяне на функции (задачи от високо ниво, специфицирани в изискванията към системата)

❖ Оперативен профил

- Операции от ниско ниво, реализирани в системата

Процедура за разработване: Муса-2

❖ Подходяща за малки системи с хомогенни потребители

❖ Стъпки на разработване

- Идентифициране на инициаторите на операциите
 - ✓ Потребители, друг софтуер, хардуер, мрежа и др.
- Избор на таблично или графично представяне
- Създаване на списък с операции
 - ✓ Създават се списъци за всеки един тип инициатор, след което списъците се обединяват
- Определяне на честотата на използване на операциите
 - ✓ Обикновено се използва действително измерване
- Определяне на вероятността на използване на операциите
 - ✓ При липса на действителни измервания се използват проучвания, експертни оценки и др., при което предходната стъпка може да се пропусне

$$0 \leq p_i \leq 1, \quad \sum_i p_i = 1$$

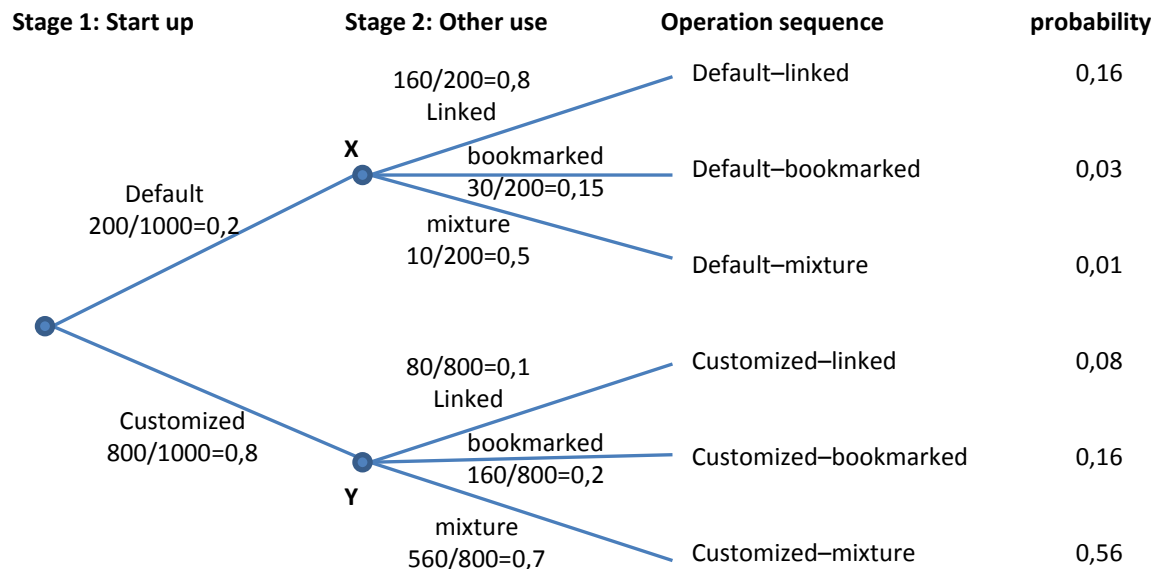
Представяне на оперативен профил с дърво

❖ Конструирание на дървото

- Последователността на операциите се разделя на фази с фиксиран брой възможности за избор
- Дъгите в дървото се асоциират с операции, за които са отбелязани честотата и вероятността на използване

❖ Пример: моделиране на използването на уеб с двуфазен процес

- Фаза 1. Стартиране на браузър с две възможности: (1) отваряне на страница по подразбиране или (2) отваряне на страница, зададена от потребителя
- Фаза 2. Последващо използване с три възможности: (1) отваряне на линкове в посетени страници, (2) отваряне на запаметени страници или (3) комбинация от двете



Софтуер за зареждане на пилотни мисии в авиацията

CASE STUDY

Контекст на софтуерния продукт

❖ Предназначение на софтуера

- Зареждане на информация за планирани мисии върху устройство за четене и запис
- Устройството се използва от пилоти за зареждане на данни в авионика (електронна техника в самолетите)
- Софтуерът предоставя графичен потребителски интерфейс за планиране на пилотни мисии

❖ Разработчици на оперативния профил

- Софтуерен продуктов мениджър
 - ✓ Продуктово планиране и маркетингова дейност
- Софтуерни инженери по тестване
 - ✓ Осигуряване на информация за тестовите входни състояния за типичните операции
- Системни инженери
 - ✓ Специфициране на системните изисквания, проектиране на високо ниво и осигуряване на функции, които се верифицират по време на тестване

Потребителски профил и системни режими

- ❖ Клиентският профил не е релевантен
- ❖ Потребители на софтуера
 - Самолетни пилоти: основни потребители
 - Тестери на полети: подпомагат самолетните пилоти при планиране на пилотни мисии
 - Тестери на авионика: извършват системна тестове на авиониката
 - Системни администратори: дейностите от тази група се извършват от самолетните пилоти
- ❖ Системни режими
 - Планиране на мисия преди полет
 - Системно тестване на авиониката
 - Администриране на системата
- ❖ Обединяване на функционалния и оперативния профили

Потребителски профил

User group	Marketing concerns	Frequency of use	Total weighting factor
Air force pilot	0,85	0,05	0,45
Flight Test support	0,10	0,80	0,45
Avionic system test	0,05	0,15	0,10

Оперативен профил*

- ❖ Прилага се имплицитен функционален профил Муса-1
 - Всяка дейност се асоциира със собствен диалогов прозорец или екран
 - Потребителят извършва една или повече дейности и приключва сесията
 - Редът на дейностите не е фикциран
- ❖ Създават се индивидуални оперативни профили за потребителските групи, които се обединяват в общ оперативен профил
- ❖ Операциите се разпределя в категории според използваемостта им

High	Medium-high	Medium-low	Low
DTC Load Inventory Save Route Planning Print	DTC Read Delete Retrieve Route	Wpn Prof Hot Keys Comm Retr/Save SCL Help Base Default FCR Mstr Mode	Retr Canned Save Canned DTC Test
High usage = 100% - 75%	Medium-high usage = 74,9% - 50%	Medium-low usage = 49,9% - 25%	Low usage = 24,9% - 0%

**Стъпките по разработване на функционален и оперативен профил се обединяват поради незначителната разлика между проектираните функции на високо ниво и реализираните операции на ниско ниво*

Метрики, валидация на резултати и изводи

- ❖ Интервюта с продуктовите мениджъри
 - Събиране на информация за изискванията за потребителския и функционалния профил
- ❖ Консултации със системните инженери и инженерите по тестване
 - Подготовка на проучване сред потребителите
- ❖ Провеждане на проучване
 - Очаквани участници – 30-50, действителни участници – 12
- ❖ Индивидуални интервюта със софтуерния екип за интерпретиране на оперативния профил
 - Съпоставяне на текущите тестови стратегии с идентифицираните нужди на клиентите
- ❖ Преглед и оценка на оперативния профил от продуктовия мениджър, системните инженери и инженерите по тестване
 - Резултатите за “high” операциите са очаквани
 - Класификацията на операцията Hot Keys като “medium” и на операцията “Help” като “medium-low” е неочакван резултат
- ❖ Резултати
 - Подобряване на тестовата стратегия
 - Влияние върху системното проектиране и проектирането на високо ниво
 - Подпомагане на взимането на решения и следващите дейности

