



Testarea Black-Box

Rusu Cristian

Scopul: identificarea conditiilor de test si crearea cazurilor de test.

Etapele de proiectare:

- Ce se testeaza?
- Specificatii pentru cazurile de test
- Procedura de testare

Tehnici de proiectare:

- Black-box (bazate pe specificatii, experienta)
- White-box (structurale)

Caracteristici:

- se folosesc modele, formal sau informal, pentru descrierea softwareului sau a unei componente a acestuia
- din aceste modele se extrag sistematic cazurile de test
- experienta oamenilor este utilizata pentru a crea cazuri de test : testerii, dezvoltatori, utilizatori
- “error guessing” : informatia despre posibile defecte si distributia acestora ajuta la proiectarea cazurilor de test

Tehnici:

- Equivalence partitioning
- Boundary value analysis
- Testarea folosind tabele de decizie
- Testarea folosind tranzitia starilor
- Testarea folosind cazuri de utilizare

Equivalence partitioning (EP)

- Intrările sistemului sau al software-ului sunt împartite în grupuri pentru care se așteaptă un comportament similar
- Identificarea de partitii echivalente (sau clase) pentru:
 1. date valide
 2. date invalide
- Se aplica și pentru: ieșiri, valori interne, valori cu referire la timp (ce se modifică înainte sau după apariția unui eveniment), parametri ai interfeței.
- EP utilizată la toate nivelele de testare

Equivalence partitioning (EP)

- Un interval continuu de valori va avea o clasa de valori valide si doua clase de valori invalide
- Exemplu:

“Goofy Mortgage Company (GMC) acorda un credit ipotecar pentru persoane cu venituri intre \$1,000/luna si \$83,333/luna. Orice venit sub \$1,000/luna nu se califica. Orice venit peste \$83,333/luna este mult prea mare, beneficiarul neavand nevoie de serviciile companiei.”

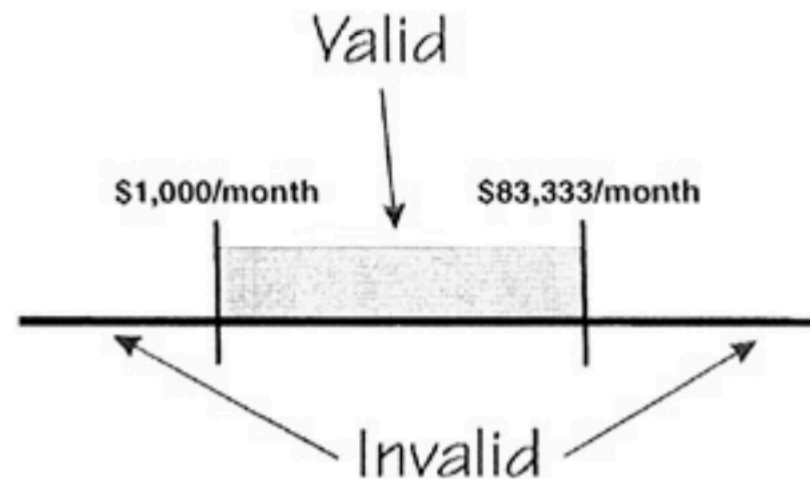


Fig. 1 Clasa continua de echivalente

Equivalence partitioning (EP)

- Daca o conditie de intrare ia valori discrete intr-un interval de valori permise exista o clasa de valori valide si doua clase de valori invalide
- Exemplu:
“GMC acorda credit ipotecar de la una pana la 5 case. Zero sau mai putine case nu reprezinta o intrare valida, nici sase sau mai multe. Nici valorile fractionare sau cu zecimale nu trebuie acceptate.”

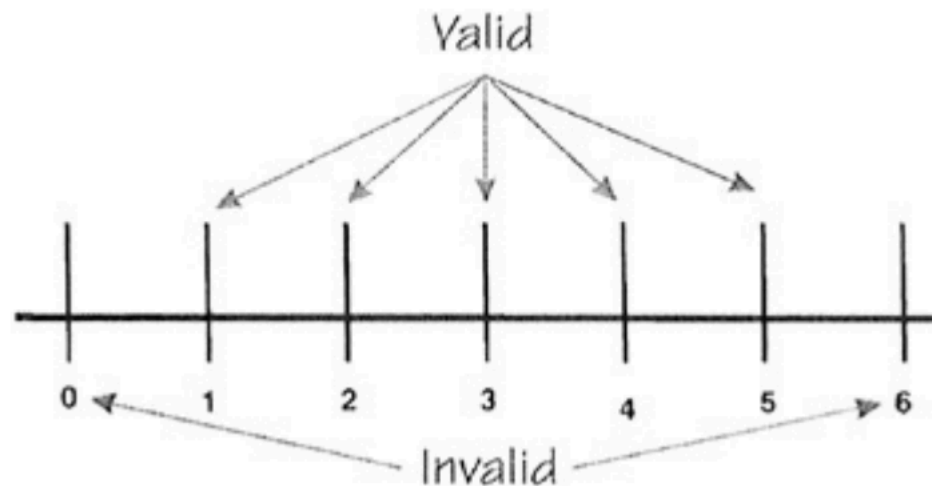


Fig. 2 Clase de echivalente discrete

Boundary Value Analysis

- Se analizeaza comportamentul la marginea fiecarei partitii de echivalenta

Definitii:

Boundary values = valoarea maxima si valoarea minima ale unei partitii

Valid boundary value = boundary value pentru o partitie valida
Invalid boundary value = boundary value pentru o partitie invalida

OBS:

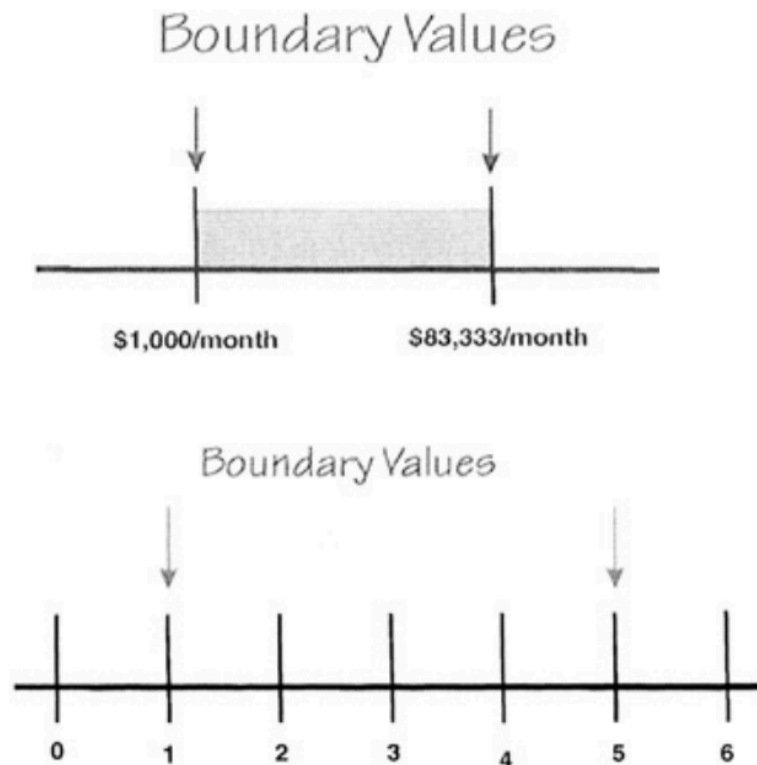
Testele trebuie create atat pentru valorile limita valide cat si invalide!

Tehnica poate fi aplicata la toate nivele de testare.

Boundary Value Analysis

- Exemplu:

“Goofy Mortgage Company (GMC) acorda un credit ipotecar pentru persoane cu venituri intre \$1,000/luna si \$83,333/luna. GMC acorda credit ipotecar de la una pana la 5 case. “



C

Conditia 1:

Date de intrare pentru limita inferioara:
{999\$, 1000\$, 1001\$}

Date de intrare pentru limita superioara:
{83332\$, 83333\$, 83334\$}

Conditia 2:

Date de intrare pentru limita inferioara:
{0, 1, 2}

Date de intrare pentru limita superioara:
{4, 5, 6}

Boundary Value Analysis

Exercitiu:

“O banca acorda credite persoanelor intre 18 si 64 ani, inclusiv. Campul de intrare corespunzator ‘Varsta’ accepta numai numere pozitive, 2 digiti.

Care sunt valorile valide si invalide pentru Equivalence Partitions si Boundary Values? “

Equivalence partitioning

Tip	Invalid	Valid	Invalid
Partitie	<17	18 – 64	65 – 100
Valoare	9	42	82
Rezultat asteptat	Rejectat	Acceptat	Rejectat

Boundary Value Analysis

Tip	I	V	V	V	V	I
Boundary	18			64		
Valoare	17	18	19	63	64	65
Rezultat asteptat	R	A	A	A	A	R

Testare cu tabele de decizie

- Utile pentru a acoperi specificatiile ce contin conditii logice si pentru a documenta structura interna a sistemului
- Conditile de intrare si actiunile sunt enuntate a.i. pot fi adevarate sau false
- Tabelul de decizie:
 1. Coloanele definesc reguli “business” i.e. combinatii de conditii ce rezulta in actiuni asociate regulilor
 2. Liniile definesc conditii si actiuni
- OBS: Trebuie sa se realizeze cel putin un test pentru fiecare coloana!

Punctul forte: creeaza combinatii de conditii care poate nu au fost exersate in timpul testarii.

Testare cu tabele de decizie

Exemplu:

Se da tabelul de decizie:

	<u>Regula 1</u>	<u>Regula 2</u>	<u>Regula 3</u>	<u>Regula 4</u>
<u>Conditii</u>				
<u>Rezident in UK?</u>	F	T	T	T
<u>Varsta intre 18-55 ani?</u>	<u>Nu conteaza</u>	F	T	T
<u>Fumator?</u>	<u>Nu conteaza</u>	<u>Nu conteaza</u>	F	T
<u>Actiuni</u>				
<u>Client asigurat?</u>	F	F	T	T
<u>Se ofera 10% reducere?</u>	F	F	T	F

1. Jack este un fumator in varsta de 32 ani rezident in Londra.
2. Michel este un nefumator rezident in Paris.
3. Fred este un tanar de 25 ani rezident in Manchester nefumator.
4. Alice este o tanara de 17 ani din Southampton fumatoare.

Testare prin tranzitia starilor

- Sistemul reprezentat printr-o diagrama de tranzitie a starilor
- Starile sistemului sunt finite si identificabile
- Tabelul de stari:
 1. relatia dintre stari si evenimente
 2. tranzitii valide si invalide

	A	B	C	D	E	F	G
SS	S1	-	-	-	-	-	-
S1	-	S2	S3	-	-	-	-
S2	-	-	-	S4	-	-	-
S3	-	-	-	-	S4	ES	-
S4	-	-	-	-	-	-	S1
ES	-	-	-	-	-	-	-

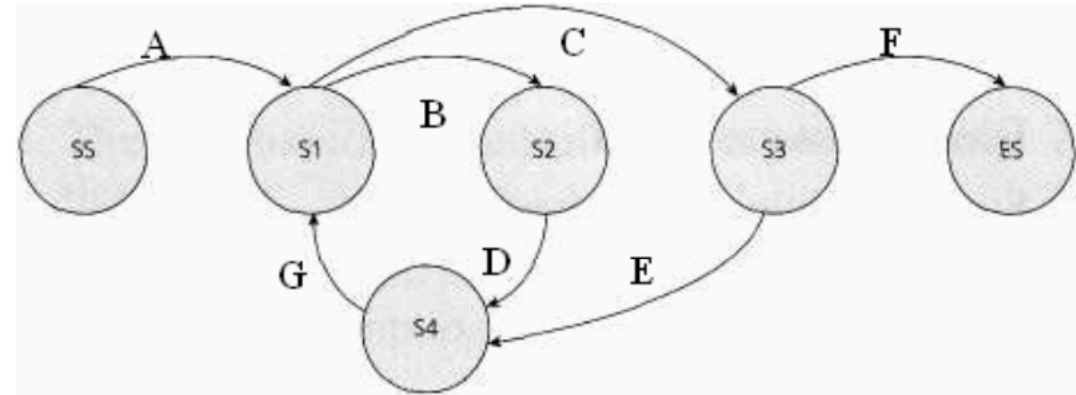


Diagrama de stari

- Utilizat in industria de embedded software si in testarea automata

Testarea cazurilor de utilizare

- Teste specificate in cazuri de utilizare sau scenarii business
- Un caz de utilizare descrie interactiunea intre actori, utilizatori si sistem, ce produce un rezultat cu o anumita valoare pentru utilizator
- Fiecare caz de test :
 1. Preconditii
 2. Postconditii
- Utilizata in cadrul testare de acceptanta si de integrare