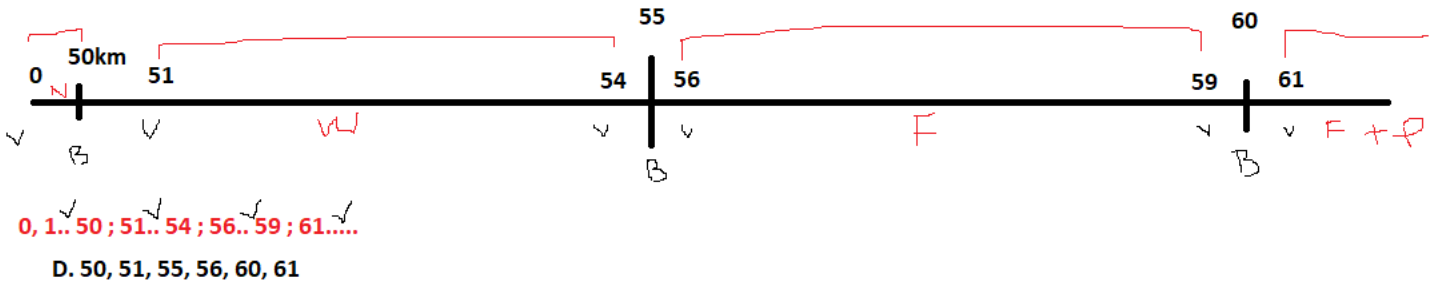
Black Box test design techniques practice. Pt.1: equivalence partitioning, boundary value analysis, pairwise testing

Assignment

Stefania Pruteanu

LEVEL 1. The speed control system has the following characteristics:



Answer is: D. 50, 51, 55, 56, 60, 61

I can't imagine the horrors of the driver who's going at 60 and trying not to slow down or speed up :))))

LEVEL 2. A device that measures the time and intensity of sunlight received by a plant counts a combination of parameters:

TESTS / CONDITIONS	**T1**	**T2**	**T3**
HOURS	1,5 (<3)	7 (6<)	0,5 (<3)
INTENSITY	very low (1)	medium (3)	very low (1)

What is the **minimum** number of **additional test cases** required to ensure that **all valid equivalence classes** are covered?

Answer is: B. 2 ;

TESTS / CONDITIONS	T1	TX	T2	T3
HOURS	1,5 (<3)	X (3-6)	7 (6<)	0,5 (<3)
INTENSITY	very low	high (4), medium, low (2), very low	medium	very low

For a minimum number from each partition missed by tests :

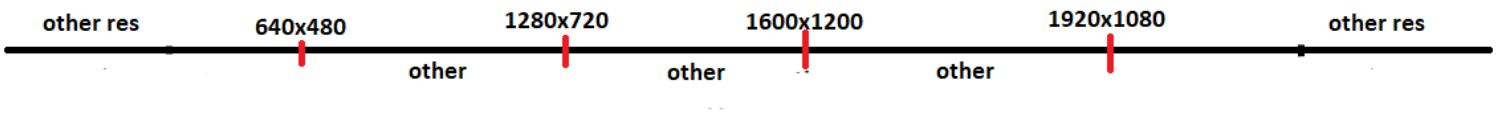
3-6 hrs - Tx - low = 1 +

6< hrs - T2 - high = 1 +

TOTAL min. Tests needed = 2.

TESTS / CONDITIONS	T4	T5
HOURS	5 (3-6)	9 (6<)
INTENSITY	low	high

2. The video playback application has requirements. The application will work on devices with this resolution:



So if any other resolution will not work

Then we treat the working resolutions as unique partitions

And the requirement says they're all valid, so just 1 test for each - 4x1 = 4 total tests

Answer is: C. Check that the application plays video on displays of all sizes specified in the requirements (4 test cases)

LEVEL 3.

Write requirements for the app that would regulate the minimum and maximum size of photos the users can upload to the system.

Also, mention the following parameters: **minimum length of comments under photos, maximum length of comments under photos** Write test cases that incorporate equivalence partitioning and boundary value analysis and will allow you to verify these requirements.

REQ:

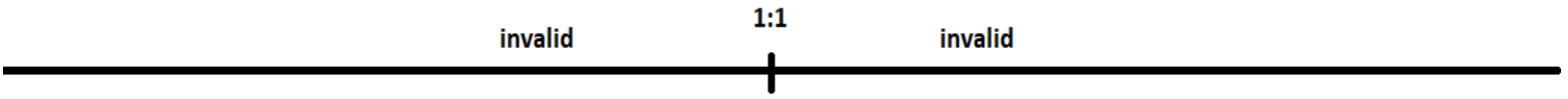
1. **For any of JPG and PNG image types, with the aspect ratio of 1:1:**

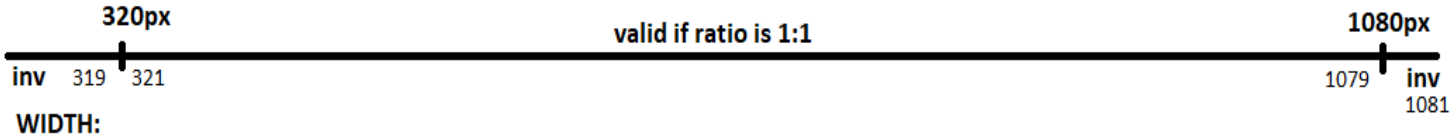
   The system accepts a width size within 320 min. and 1080 max. pixels.

   The system accepts a height between 320 min. and 1080 max pixels.

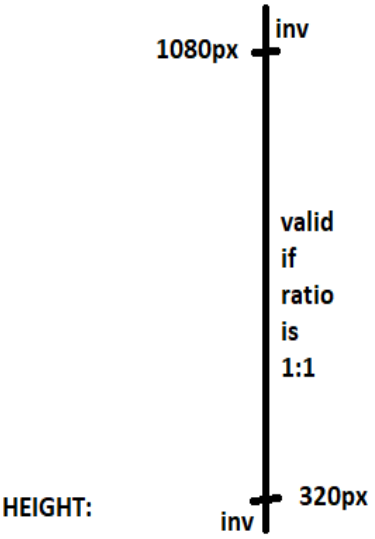
I will start with the BVA here: //// **Also I made the test cases in TestRail under the project**

ASPECT RATIO:





WIDTH:



So we have 6 Boundary Values that will generate 36 matches for width and height - 6^2.

The size range killed my brain, it can be too low, valid or too high for BOTH **W** and **H** (=**12**)

The 12 matches are either **valid or invalid** for “1:1” ratio = **12:2**

Remaining 6 matches are assigned to each BV (319, **320, 321, 1079, 1080**, 1081):

4 values represent 1 valid class, 2 values represent the same 1 invalid class

IMG	WIDTH	HEIGHT	> >	SIZE	RATIO	UPLOAD
JPG	320	320		OK	1:1	[ALLOWED]
JPG	321	319		-	.	-
JPG	1079	1081		-	.	-
JPG	1081	319		-	.	-
JPG	1079	319		-	.	-
JPG	1081	321		-	.	-
JPG	1080	319		-	.	-
JPG	1080	1079		OK	.	-
JPG	321	1079		OK	.	-
JPG	1080	1081		-	.	-
JPG	1079	320		OK	.	-
JPG	1081	1081		-	1:1	-
JPG	1080	1080		OK	1:1	[ALLOWED]
JPG	1081	1079		-	.	-
JPG	1080	320		OK	.	-
JPG	1079	1079		OK	1:1	[ALLOWED]
JPG	321	1080		OK	.	-
JPG	320	1080		OK	.	-
JPG	1079	321		OK	.	-
JPG	319	319		-	1:1	-
JPG	1079	1080		OK	.	-
JPG	1080	321		OK	.	-
JPG	320	1081		-	.	-
JPG	321	321		OK	1:1	[ALLOWED]
JPG	320	321		OK	.	-
JPG	319	1081		-	.	-
JPG	321	1081		-	.	-
JPG	321	320		OK	.	-
JPG	319	1079		-	.	-
JPG	319	321		-	.	-
JPG	319	320		-	.	-
JPG	319	1080		-	.	-
JPG	1081	1080		-	.	-
JPG	320	319		-	.	-
JPG	320	1079		OK	.	-
JPG	1081	320		-	.	-

```
################################################################
# Paste test factors here.
# Check the documents for more details.
# https://github.com/Microsoft/pict/blob/master/doc/pict.md
################################################################

IMG:            JPG
WIDTH:          319, 320, 321, 1079, 1080, 1081
HEIGHT:         319, 320, 321, 1079, 1080, 1081
> >:
SIZE:           OK, -

RATIO:          1:1, .

UPLOAD:         [ALLOWED], -

IF
([WIDTH] >= 320 AND [HEIGHT] >= 320) AND ([WIDTH] <= 1080 AND [HEIGHT] <= 1080)
THEN [SIZE] = "OK"
ELSE [SIZE] = "-";

IF ([WIDTH] = [HEIGHT])
THEN [RATIO] = "1:1"
ELSE [RATIO]  = ".";

IF [SIZE] = "OK" AND [RATIO]= "1:1"
THEN [UPLOAD] = "[ALLOWED]"
ELSE [UPLOAD] = "-";
```

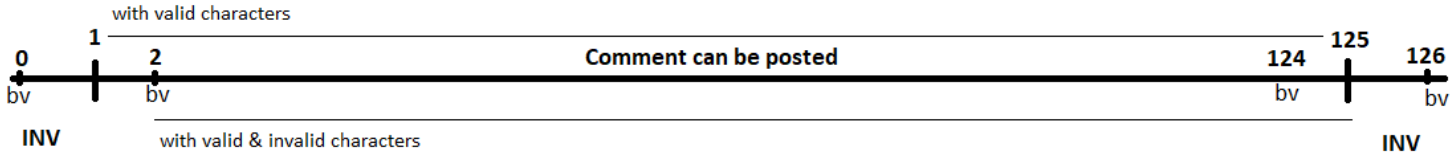
^ This is how I generated the pairwise table from https://pairwise.yuuniworks.com/

Export from **TestRail-Cat photo sharing app EP, BVA.pdf** with the tests

REQ:

2. System allows users to post comments, under any photo posted on the feed, of minimum 1 character long and maximum of 125 characters long. Comments of 1 character don't allow invalid characters.

BVA:



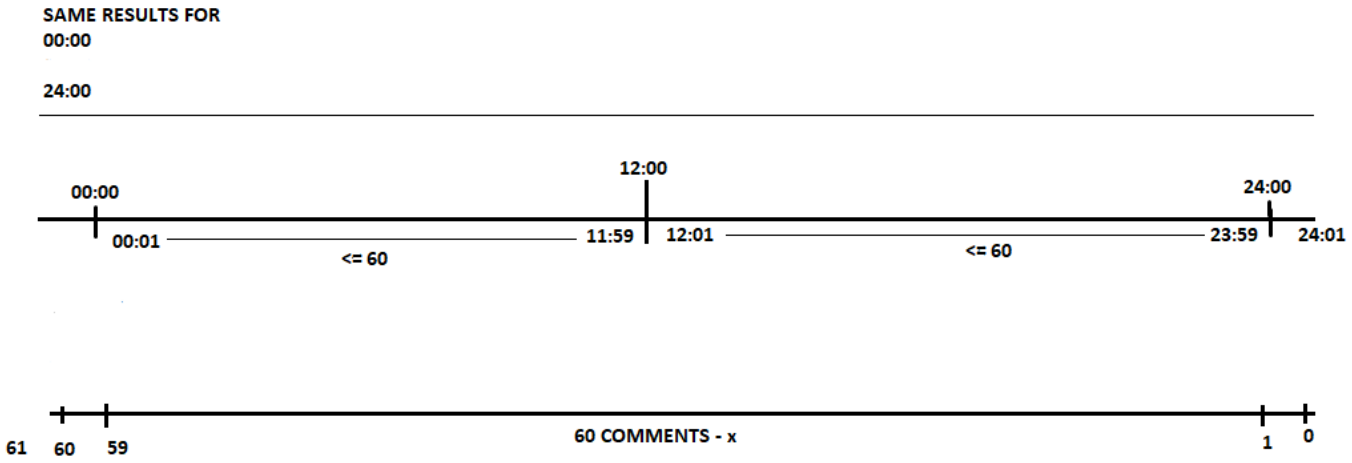
TESTS:

Is this a decision table?

TESTS / CONDITIONS	T1	T2	T3	T4	T5	T6
CHARACTERS	x	INVALID	VALID	INVALID	VALID	VALID
LENGTH	***0***	***1***	***1***	***2***	***125***	***126***
STATUS:	**NOT POSTED**	**NOT POSTED**	*POSTED*	*POSTED*	*POSTED*	**NOT POSTED**

REQ:

3. System limits users to post **more than 60 comments**, under any photo posted on the feed, within **the span of 12 hours from the first comment** posted at the current time and date.



TESTS:

TESTS / CONDITIONS	T1	T2	T3	T4	T5	T6
NR COMMENTS	**61**	**60**	**59**	**1**	**0**	61
HOURS SPAN	*00:00*	***00:01***	***11:59***	***12:00***	***12:01***	*23:59*
STATUS:	**NOT POSTED**	POSTED	*POSTED*	***POSTED***	***NOT POSTED***	NOT POSTED