

For all of the cases below, do the traditional equivalence class and boundary analysis. Draw one table and use a new line for each variable. There is at least one variable of interest in each case below. It's up to you to identify the variable(s) to analyze. Along with the table, provide one or more pages of notes to explain any aspect of your paper that needs an explanation.

You may work on this in pairs. If both of you are stuck, you may consult with another pair. You may use any reference material. However, each of you must think through, and be able to explain, every entry in your pair's table.

Overall, I expect a clearly labelled, well-organized, readable table. These tables summarize a lot of information. They have little value if they don't present that information in a straightforward, simple way.

The traditional table has the following columns:

- *Variable name*
- *Valid equivalence class*
- *Invalid equivalence classes*
- *Boundary values to test*

The table often includes special cases in the boundary values column, but it is essential that the boundary values be presented first, and that it be clear what values are bounds and what are other special values.

Such tables also often include a Notes column.

In this assignment (and in practice) I also recommend that you provide additional explanatory documentation on a separate page, outside of the constraints of the table.

For example, in Question 4, you are either obtaining the boundary values by reference to documentation (what documents?) or by working with the program to see what it accepts (whether those values are correct or not). It might take a few lines of text to explain what you actually did. This is too much for a table.

1. You are given a VISA gift card, with a \$500.00 balance. You go to a store, buy some merchandise, and the clerk enters a total amount to be deducted from your card.

This is the most straightforward analysis.

The valid range is \$0.00 to \$500.00.

Some students would say that \$0.00 is an invalid sale price. For them, the smallest valid is \$0.01.

Out of bounds values are \$-0.01 and \$5.01.

501 is a common error, as is -1. Both ignore the values after the decimal point.

2. You are required to enter a password of up to 100 characters. The characters must be lower ASCII, printing characters.

You can analyze this in two fundamentally different ways:

- *As a field with a length of 0 to 100 characters*
- *As a field whose characters must be printing lower ASCII characters (see <http://www.columbia.edu/kermit/ascii.html> for an ASCII table)*

*It is important to keep these analyses **separate**. Don't confuse your reader by treating them together. Use one line of the table for field length. Use **another line** for character set.*

From there, the analysis is straightforward:

- *Length: Valid 0-100; Invalid <0? (impossible to have a negative length string) and >100 characters. Boundaries at 0, 100 and 101.*
- *Character: valid codes from 32 (space) through 126 (~). Invalid codes below (0 to 31) and above (127 plus).*

3. People are divided into three teams, Red, Green, Blue. You enter the person's name and their team. (Analyze Team, not Name).

*Analyze the Team (red green blue) and **NOT** the person's name. On an assignment, and possibly on a midterm or final exam, I take points off when someone adds complexity to their answer that they were specifically told not to add.*

There are 3 valid values, Red, Green and Blue. Anything else is invalid. There are no boundaries because there is no differentiation between members of the team. All Reds are the same, so any Red will do as a test case. There are no boundaries of the invalid class, nor any obvious

"best representatives". This illustrates why I said in lecture that domain analysis is inappropriate for enumerated variables.

4. Go to the page setup dialog in Open Office. Analyze the variable that specifies page width.

Students typically solved this by going to the Page Style dialog and seeing what the program accepts.

Problem: If the program has a boundary-related error, relying on what the program tells you are its boundaries will give you the wrong answer.

That said, the max appears to be 46.85" tall or wide and the program appears to round values entered with more than 2 digits after the decimal.

If the upper bound is 46.85, the value that is too big is 46.855, the smallest value that should round up, to 46.86.

5. Go to the page setup dialog in Open Office. Analyze the variable that specifies number of copies to print.

The largest number accepted in the Print dialog (OO 1.9.77) is 9,999. This is a typical integer numeric input field, whose lower boundary (invalid) might be 0 or -1.

6. A and B and C are unsigned integers. $C = A+B$. Analyze C.

C ranges from 0 to MaxInt. It should be impossible to load any other values into C. Invalid boundaries are -1 and MaxInt+1

7. A and B and C are unsigned integers. $C = A+B$. Analyze the pairs (A,B) that yield valid values of C.

We've done the analysis for C. Suppose the interesting values for C are -1, 0, 1, MaxInt and MaxInt+1. We need to find (A,B) pairs that would yield these values for C. For example, there are no values that we can input into (A,B) that can yield $A+B < 0$. One value generates a sum of 0. Two values (0,1) and (1,0) generate a 1. Note that if A is an integer, so is $A/2$, and if A is odd, the result is truncated. MaxInt is generally $2^N - 1$,

for some value of N.

8. You are required to do a certain task on July 1, 2005. On completion of the task, you enter the time (in hours, minutes and seconds) at which you completed the task. Suppose that you do in fact complete the task on July 1, 2005. For simplicity, use military time (1 p.m. is 1300 hours).

Valid values range from 00:00:00 through 23:59:59

9. X is a floating point number, stored to 5 decimal digits of precision. Input a value to X that should be equivalent to PI.

With 5 digits of precision, PI is 3.1416. 1000 PI is 3141.6. And PI/1000 is 0.0031416. It is common for students to confuse "fixed" formats (N digits after the decimal) with precision (how many significant digits).

A value 3.14161 should be equivalent to PI, so should 3.14159. In general, any value from 3.141550 to 3.141649 should be equivalent to PI. Those are the upper and lower bounds of the equivalence class. (Should you use 6 digits? 7? That depends on what the program will accept / interpret as an input value without rejecting it as too long.

10. X is a floating point number, stored to 5 decimal digits of precision. Input a value to X that should be equivalent to PI /10000.

The analysis of this is the same as for question 9, except that it explicitly checks whether the student understands what decimal precision means.

Understanding the notion of decimal precision is critical for deciding whether a complex calculation was correct or not. A calculated value can be "wrong" but within the tolerance limits implicit in limited-precision arithmetic. In that case, the result is typically interpreted as correct.

CLASSROOM NOTES:

Students worked on this assignment in pairs, but when one pair got stuck, they were free to consult with other pairs or with online material or books. Note that I am asking for the traditional table (Myers) not the risk-driven table. The goal for this assignment is to make sure students are familiar with the traditional analysis and have applied it to

a range of problems.

These problems are different enough from each other that different students needed significantly different coaching.