Berkeley Math Circle

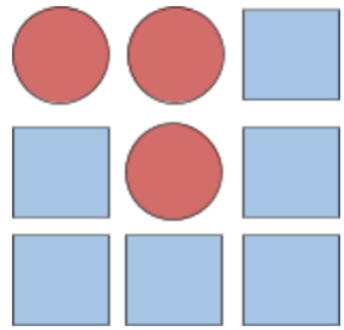
Information, Number Systems, and Computers

Circle-Square Patterns

Challenge 1: Patterns

Record all of the possible three place value patterns that use only the circle and square shapes. A few are listed in the diagram to the right.

Try to make sure these patterns are in some type of order and avoid putting down patterns randomly



1) ____ ____ ____

2) ____ ____ ____

3) ____ ____ ____

4) ____ ____ ____

5) ____ ____ ____

6) ____ ____ ____

7) ____ ____ ____

8) ____ ____ ____

Challenge 2: Describe Your Rules

Reflect on how you put the patterns above in order from one line to the next. Are there any clear rules to follow? If you didn't use any clear rules, re-do Challenge 1 with some rules in mind.

Describe the rules for how you listed your patterns above. Your rules should clearly describe how to create the *exact same* list of patterns.

____

____

____

____

____

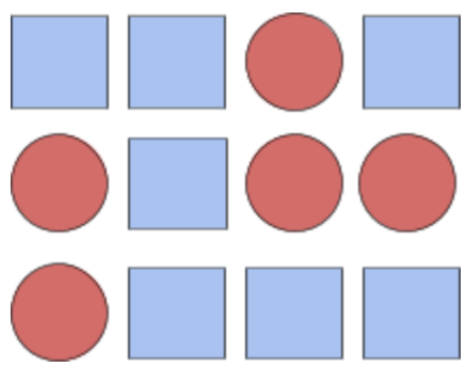
____

____

____

____

Challenge 3: More Patterns

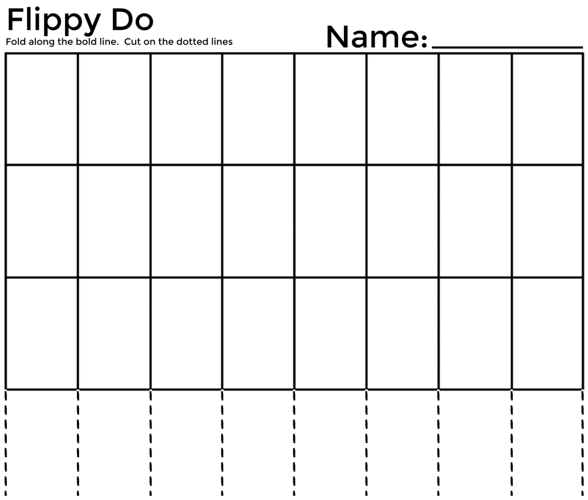


Use your rules from the last page to try and generate all possible four place value patterns using only circles and squares. Three examples are listed to the right.

You may need to add new rules or slightly change your rules to account for all four place value patterns, but try to keep them as similar as possible.

Flippy-Do

On a separate sheet of paper, draw 7 vertical lines (folding in half three times, "hamburger style") and 3 horizontal lines (folding in half twice "hotdog style") to make your flippy do. It should look something like this graphic below.



Use your flippy-do to answer the following questions.

All 4-Bit Numbers: Fill in the binary equivalents for the decimal numbers below. We've started the first three for you.

Binary: 4-bit number	Decimal
0000	0
0001	1
0010	2

Binary: 4-bit number	Decimal

What do you notice when you compare the odd numbers with the even numbers? What might explain this?

Binary Numbers with exactly one 1: Complete the chart with all 8-bit binary numbers that have exactly one 1. We've done the first two for you.

Binary: 8-bit number (with exactly one *1*)	**Decimal**	**Binary: 8-bit number** (with exactly one *1*)	**Decimal**
0000 0001	1		
0000 0010	2		
0000 0100			

What do you notice about the decimal equivalents above?

Conversion Practice: Find the equivalent binary or decimal numbers below.

Binary	**Decimal**	**Binary**	**Decimal**
100			5
101			17
1101			63
0001 1111			64
0010 0000			127
1010 1010			256
1111 1111			513

When you add a zero to the right of a decimal number, it multiplies its value by 10 (For example, 15 becomes 150). What similar result happens to the value of a binary number when you add a zero on the right? (For example, 11 would become 110).

Do the binary numbers 0011 and 000011 have the same value or different values? Explain.

Would two bits be enough to assign a unique binary number to each vowel in the English language? Explain.

How many bits would you need if you wanted to count up to the decimal number 1000?