BASE CONVERSION

COP 3502
The Decimal Numbering System

- What is the decimal numbering system?
- Have you ever thought about why we use this system or how it came about?
Our regular counting system is decimal
- i.e. Base 10
  - This is because we use 10 distinct digits, 0 \(\rightarrow\) 9
- In general the numerical value of a number, is what you were taught in elementary school:
  - 2713
  - \(= 2 \times 10^3 + 7 \times 10^2 + 1 \times 10^1 + 3 \times 10^0\)

Where each digit’s value is determined by what place it’s in, going from right to left:

3 is in the 1’s place 3
1 is in the 10’s place 10
7 is in the 100’s place 700
2 is in the 1000’s place 2000

Note: each ‘place’, is a perfect power of the base, i.e. 10

\[
2713
\]

\[
= 2 \times 10^3 + 7 \times 10^2 + 1 \times 10^1 + 3 \times 10^0
\]
Although this may seem to be the only possible numbering system,

- it turns out that the number of digits used is arbitrary!
- We could just have easily chosen 5 digits (0→4),
  - in which case the value of a number would be:
    - $314_5 = 3 \times 5^2 + 1 \times 5^1 + 4 \times 5^0 = 84_{10}$

Thus, this is how we convert from a different base to base 10
Base Conversion

- $314_5 = 3 \times 5^2 + 1 \times 5^1 + 4 \times 5^0 = 84_{10}$

- In general, we can use the following formula:

$$d_{n-1}d_{n-2}...d_2d_1d_0 \text{ (in base } b) = d_{n-1}xb^{n-1} + d_{n-2}xb^{n-2} + ... + d_2xb^2 + d_1xb + d_0$$

(Nota: $b$ raised to the 1 and 0 powers were simplified)
Base Conversion to Base 10

- Given, $781_9$ what would this be base 10?
  - $781_9 = 7 \times 9^2 + 8 \times 9^1 + 1 \times 9^0$
  - $= 640_{10}$

- Given, $1110101_2$ what would this be base 10?
  - $1110101_2 = 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$
  - $= 117_{10}$

- Base 2 is so common it’s called binary
  - (heard of it before?)
The uses of Binary and Hexadecimal

- A “digital” computer, (vs analog) operates on the principle of 2 possible states – ON and OFF
  - This corresponds to there being an electrical current present, or absent.
- So ON is “1” and OFF is “0”
- Each binary digit, or “bit” corresponds to a single “switch” in a circuit
- And if we add up enough switches we can represent more numbers, so instead of one digit we can get 8 to make a byte.
So why do all computers use binary then?

Simple answer – computer weren’t designed to use binary, rather, binary was determined to be the most practical system to use with the computers we did design.
So this explains why we would want to use binary, but why would we ever want hexadecimal?

- Octal and hexadecimal are simply a shorter representation of binary, or a more human readable version of binary.
- Usually used more memory addresses or RGB color values
- So if you want to specify this color:  
  - You don’t have to type: 1111 1111 0000 0000 1111 1111
  - You can type: #FF00FF
Hexadecimal

- The most common base above 10 is 16
  - Which is known as Hexadecimal
  - The 16 digits are:
    - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
      - Where A = 10, B = 11, C = 12, D = 13, E = 14, F = 15
  
- Note that converting from Hexadecimal to Decimal, is the same as the previous slide

- $A3D_{16} =$
  - $= Ax16^2 + 3x16^1 + D = 10x16^2 + 3x16 + 13$
  - $= 2621_{10}$
Converting from Hexadecimal to Binary

- Since 16 is a perfect power of 2, converting to base 2 is relatively easy 😊
  - We find that each hexadecimal digit is perfectly represented by 4 binary digits (since $16 = 2^4$)
- Here’s a chart with the conversions between each hexadecimal digit and base 2:

<table>
<thead>
<tr>
<th>Hex</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bin</td>
<td>0000</td>
<td>0001</td>
<td>0010</td>
<td>0011</td>
<td>0100</td>
<td>0101</td>
<td>0110</td>
<td>0111</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Hex</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bin</td>
<td>1000</td>
<td>1001</td>
<td>1010</td>
<td>1011</td>
<td>1100</td>
<td>1101</td>
<td>1110</td>
<td>1111</td>
</tr>
</tbody>
</table>
Using the chart below,

What is $\text{ACD}_{16}$ in binary?

\[ = 1010\ 0011\ 1101_2 \]

<table>
<thead>
<tr>
<th>Hex</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bin</td>
<td>0000</td>
<td>0001</td>
<td>0010</td>
<td>0011</td>
<td>0100</td>
<td>0101</td>
<td>0110</td>
<td>0111</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Hex</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bin</td>
<td>1000</td>
<td>1001</td>
<td>1010</td>
<td>1011</td>
<td>1100</td>
<td>1101</td>
<td>1110</td>
<td>1111</td>
</tr>
</tbody>
</table>
Let’s consider converting a number in decimal to another base, say binary

For example convert $117_{10}$ to binary

We know that the format is going to look something like this:

$$117_{10} = d_6 \cdot 2^6 + d_5 \cdot 2^5 + d_4 \cdot 2^4 + d_3 \cdot 2^3 + d_2 \cdot 2^2 + d_1 \cdot 2^1 + d_0 \cdot 2^0$$

So, what we know is this:

All of the terms on the right hand side are divisible by 2 except the last.

- Sooo all of them will give a remainder of 0, when divided by 2
- Thus we know that $d_0 = 117 \% 2$, this is the remainder
Converting from Decimal to Another Base

- $117_{10} = d_6x2^6 + d_5x2^5 + d_4x2^4 + d_3x2^3 + d_2x2^2 + d_1x2^1 + d_0x2^0$

- Now if we divide the right hand side by 2, using int division we get:
  - $d_6x2^5 + d_5x2^4 + d_4x2^3 + d_3x2^2 + d_2x2^1 + d_1x2^0$
  - This number must equal $117/2 = 58$,
  - So we’ve come up with a process of continually reducing the equation by a factor of 2...
Converting from Decimal to Another Base

- \[ 117_{10} = d_6 \times 2^6 + d_5 \times 2^5 + d_4 \times 2^4 + d_3 \times 2^3 + d_2 \times 2^2 + d_1 \times 2^1 + d_0 \times 2^0 \]

Here is the whole process:

- \[ 117 \mod 2 = 1 \ (d_0) \quad 117/2 = 58 \]
- \[ 58 \mod 2 = 0 \ (d_1) \quad 58/2 = 29 \]
- \[ 29 \mod 2 = 1 \ (d_2) \quad 29/2 = 14 \]
- \[ 14 \mod 2 = 0 \ (d_3) \quad 14/2 = 7 \]
- \[ 7 \mod 2 = 1 \ (d_4) \quad 7/2 = 3 \]
- \[ 3 \mod 2 = 1 \ (d_5) \quad 3/2 = 1 \]
- \[ 1 \mod 2 = 1 \ (d_6) \quad 1/2 = 0 \ (we
can
cstop) \]

Now, read the answer from the bottom up!
Converting from Decimal to Another Base

- $117_{10} = d_6x2^6 + d_5x2^5 + d_4x2^4 + d_3x2^3 + d_2x2^2 + d_1x2^1 + d_0x2^0$

Here is the whole process:

- $117 \% 2 = 1 \ (d_0) \ 117/2 = 58$
- $58 \ % 2 = 0 \ (d_1) \ 58/2 \ = 29$
- $29 \ % 2 = 1 \ (d_2) \ 29/2 \ = 14$
- $14 \ % 2 = 0 \ (d_3) \ 14/2 \ = 7$
- $7 \ % 2 = 1 \ (d_4) \ 7/2 \ = 3$
- $3 \ % 2 = 1 \ (d_5) \ 3/2 \ = 1$
- $1 \ % 2 = 1 \ (d_6) \ 1/2 \ = 0 \ (we \ can \ stop)$

Now, read the answer from the bottom up! 1 1 1 0 1 0 1
Practice Problem

- Convert the following from decimal to the given base:

175\textsubscript{10} = _____\textsubscript{3}

175 \mod 3 = 1 \quad 175/3 = 58
58 \mod 3 = 1 \quad 58/3 = 19
19 \mod 3 = 1 \quad 19/3 = 6
6 \mod 3 = 0 \quad 6/3 = 2
2 \mod 3 = 2 \quad 2/3 = 0,

so 175\textsubscript{10} = 20111\textsubscript{3}.

381\textsubscript{10} = _____\textsubscript{16}

381 \mod 16 = 13 (D) \quad 381/16 = 23
23 \mod 16 = 7 \quad 23/16 = 1
1 \mod 16 = 1 \quad 1/16 = 0,

so 381\textsubscript{10} = 17D\textsubscript{16}.
Converting from any base (B1) to and other base (B2) where neither base is base 10

- What do we do?
- Don’t panic!

- We can:
  1. Convert from base B1 to base 10
  2. Convert from base 10 to base B2

- Phew! We know how to do both of those!
Converting from any base

- If you are converting between two bases that are perfect powers of 2,
  - the following procedure works more quickly:
    1. Convert from base B1 to base 2
    2. Convert from base 2 to base B2
  - Consider the following example
    - We want to know what $\text{ACD}_{16}$ is in base 8
    - We know $\text{ACD}_{16} = 1010\ 0011\ 1101_{2}$ from before
    - Know we must convert the right-hand side to base 8.
Converting from any base

- We know $\text{ACD}_{16} = 1010 \ 0011 \ 1101_2$ from before.
- Now we must convert the right-hand side to base 8.
  - Remember that $8 = 2^3$, so three binary digits perfectly represent one octal (base 8) digit.
  - So let’s just group the binary digits in sets of 3, from right to left:
    - $101 \ 000 \ 111 \ 101_2$
  - And finally convert each set of three binary digits to its octal equivalent:
    - $5075_8$
- Note: only works when one base is a perfect power of the second base.
Why this works:

\[ A3D_{16} = 10 \times 16^2 + 3 \times 16^1 + 13 \times 16^0 \]

\[ = (1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0) \times 16^2 + \]
\[ (0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0) \times 16^1 + \]
\[ (1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0) \]

\[ = (1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0) \times 2^8 + \]
\[ (0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0) \times 2^4 + \]
\[ (1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0), \text{ by rewriting 16 as a power of 2.} \]

\[ = 1 \times 2^{11} + 0 \times 2^{10} + 1 \times 2^9 + 0 \times 2^8 + \]
\[ 0 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + \]
\[ 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \]

\[ = 101000111101_2. \]
Practice Problem

- Consider writing a function that takes in a number in decimal, and prints out the equivalent value in binary.
  - We can utilize what we learned about base conversion.
  - The key is as follows:
    - If we are converting 78 from base 10 to base 2, we calculate $78 \div 2 = 39$.
    - This is the LAST digit we want to print, since it’s the units digit of our answer.
    - Preceding that zero, we must take the decimal number $78 \div 2 = 39$, and convert THAT to binary. **But, this is a recursive task!!!**