Conversion Worksheet (Chapter 1)

### Start with a "base 10" (decimal) notation review.

This is the notation you've used since you started math. The rules for representing a number in base 10 can be applied to any other "base-x" notation.

1. "Base 10" (decimal) notation uses 10 digits. Enter them in order, starting with 0, below.  
   \_\_, \_\_, \_\_, \_\_, \_\_, \_\_, \_\_, \_\_, \_\_, \_\_
2. In "base 10" (decimal) notation, a digit's value is determined by its column position. The column represents *magnitude*, that is, each column represents a group of **10** raised to an exponent. For example, in the number "23," the "3" in the far-right column means 3 x 100, which is 3 x 1, or simply three. The "2" in the column to the left means 2 x 101, which is 2 x 10, or twenty. The exponent increases by one for each column as you move right to left. Enter the exponents for the two leftmost columns below (notice the pattern set in the final three columns).  
    \_ \_ 2 1 0  
    10 10 10 10 10
3. Remember that 100 is 1, 101 is 1 x 10, 102 is 1 x 10 x 10 (or 100), 103 is 1 x 10 x 10 x 10 (or 1,000), etc. What is the magnitude represented by each column in a 5-digit decimal number?  
   \_\_\_\_ \_\_\_\_ \_100\_ \_10\_ \_1\_\_
4. So the following 5-digit number:   
   52,476   
   Is the same as:   
   5 x 10,000 + 2 x 1,000 + 4 x 100 + 7 x 10 + 6 x 1

### Try base 8 (octal) notation next.

1. "Base 8" (octal) notation uses 8 digits. Enter them in order, starting with 0, below.  
   \_\_, \_\_, \_\_, \_\_, \_\_, \_\_, \_\_, \_\_ (note, the largest digit is one less than the base value)
2. In "base 8" notation, each column represents **8** raised to an exponent. As with base 10, the exponent for a column starts with 0 for the far-right column, and increases by one as you proceed to the left. Enter the exponents for the columns below.  
    \_ \_   
   8 8 80
3. Remembering that 80 is 1, 81 is 1 x 8, 82 is 1 x 8 x 8, etc., what is the magnitude represented by each column in a 3-digit octal number?  
   \_\_\_ \_\_\_ \_1\_
4. So the following 3-digit number, which is written in base 8:   
   764   
   Is the same as:   
   7 x 64 + 6 x 8 + 4 x 1
5. You can't tell by looking at the digits that the above number is in base 8, so special notation is used. Base 8 (octal) numbers are written either with a prefix of 0o (that's a zero followed by a lowercase oh) or with a subscript of 8. For example, 0o764 or 7648.
6. What is the decimal value of 7648? (Hint: just solve the expression at the end of step 4) \_\_\_\_\_\_\_
7. To convert a decimal number to octal, determine how many of each group are in the number. To convert 500 to octal, do the following: 500 / 82 = **7** with a remainder of 52; 52 ÷ 81 is **6** with a remainder of 4. 4 ÷ 1 is **4**. So 500 converted to octal is 7648.

Convert the following octal numbers to decimal notation:

28 \_\_\_\_\_ 378 \_\_\_\_\_ 428 \_\_\_\_\_ 1218 \_\_\_\_\_ 5368 \_\_\_\_\_

Convert the following decimal numbers to octal:

4 \_\_\_\_\_ 8 \_\_\_\_\_ 14 \_\_\_\_\_ 25 \_\_\_\_\_ 100 \_\_\_\_\_

### Now, try "base 16" (hexadecimal).

1. "Base 16" (hexadecimal) notation uses 16 digits. *Since there are only 10 numeric digits, hexadecimal notation uses the first 6 letters of the alphabet to represent digits for 10 through 15.* Enter the hexadecimal digits in order, starting with 0, below.  
   \_\_, \_\_, \_\_, \_\_, \_\_, \_\_, \_\_, \_\_, \_\_, \_\_, \_A\_, \_B\_, \_\_, \_\_, \_\_, \_\_,
2. In "base 16" notation, each column represents **16** raised to an exponent. Applying the same rules as earlier for determining the exponents, enter the exponents for the columns below.  
    \_ \_ \_   
   16 16 16
3. Remembering that 160 is 1, 161 is 16, etc., what are the values for each column in a 3-digit hexadecimal number?  
   \_\_\_\_ \_\_\_\_ \_1\_\_
4. So the following 2-digit number, which is written in base 16:   
   4B   
   Is the same as:   
   4 x 16 + 11 x 1 (remember, in hexadecimal notation B is a digit that represents 11)
5. You can't always tell by looking at a number if it is in base 16, so special notation is used. Base 16 (hexadecimal) numbers are written either with a prefix of 0x or with a subscript of 16. For example, 0x4B or 4B16.
6. What is the decimal value of 4B16? \_\_\_\_\_\_\_
7. To convert a decimal number to hexadecimal, determine how many of each group are in the number. To convert 75 to hexadecimal, do the following: 75 ÷ 161 = **4** with a remainder of 11; 11 ÷ 160 is 11, which in hexadecimal digits is represented by a **B**. So 75 converted to hexadecimal notation is 4B16.

Convert the following hexadecimal numbers to decimal notation:

216 \_\_\_\_\_ C16 \_\_\_\_\_ 4216 \_\_\_\_\_ A516 \_\_\_\_\_ FF16 \_\_\_\_\_

Convert the following decimal numbers to hexadecimal notation:

10 \_\_\_\_\_ 15 \_\_\_\_\_ 160 \_\_\_\_\_ 178 \_\_\_\_\_ 22 \_\_\_\_\_

### Apply the same rules to base-2 (binary) notation.

1. "Base 2" (binary) notation uses 2 digits. Enter the binary digits in order, starting with 0, below.  
   \_\_, \_\_
2. In "base 2" (binary) notation, each column represents **2** raised to an exponent. Enter the exponents for the columns below.  
    \_ \_ \_ \_ \_ \_ \_ \_   
    2 2 2 2 2 2 2 2
3. What are the values for each column in an 8-digit binary number?  
   \_\_\_\_ \_\_\_\_ \_\_\_\_ \_\_\_\_ \_\_\_\_ \_\_\_\_ \_\_\_\_ \_\_\_\_
4. So the following 8-digit number, which is written in base 2 (binary):   
   0100 1001   
   Is the same as:   
   0 \* 128 + 1 \* 64 + 0 \* 32 + 0 \* 16 + 1 \* 8 + 0 \* 4 + 0 \* 2 + 1 \* 1
5. Base 2 (binary) numbers are usually written either with a prefix of 0b or with a subscript of 2. For example, 0b111 or 1112. *Be careful! Sometimes authors believe it's obvious in context that the number is binary, in which case they drop the prefix or subscript.* You'll also often encounter binary numbers written in groups of 4, so they might start with 0. The distinction is important! In decimal, 10 means ten; in binary, it is merely two!
6. What is the decimal value of 0100 10012? \_\_\_\_\_\_\_
7. To convert a decimal number to binary, determine how many of each group are in the number. To convert 13 to binary, do the following: 13 ÷ 23 = **1** with a remainder of 5; 5 ÷ 22 = **1**, with a remainder of 1; 1 ÷ 21 is **0**, with a remainder of 1; 1 ÷ 20 is **1**. So 13 converted to binary notation is 11012.

Convert the following hexadecimal numbers to decimal notation:

00102 \_\_\_\_\_ 00002 \_\_\_\_\_ 11112 \_\_\_\_\_ 10102 \_\_\_\_\_ 01102 \_\_\_\_\_

Convert the following decimal numbers to binary:

2 \_\_\_\_ 7 \_\_\_\_ 15 \_\_\_\_ 21 \_\_\_\_ 193 \_\_\_\_

### Adding numbers of the same base

Adding numbers of the same base is the same as adding in decimal, the old "carry the one" rule for addition is applied when a column equals or exceeds the base. For example

78 + 18 = 108.

When adding numbers that are all the same base, give your answer in the same base as the numbers in the problem.

Solve the problems below (use the rest of this sheet as scratch paper if necessary):

|  |  |  |  |
| --- | --- | --- | --- |
| 00002 + 00012 = | 00012 + 00012 = | 00102 + 00012 = | 0112 + 00012 = |
| 08 + 18 = | 78 + 18 = | 108 + 18 = | 778 + 18 = |
| 016 + 116 = | F16 + 116 = | 1016 + 116 = | FF16 + 116 = |

### Adding numbers of different bases

Practically speaking, you'll rarely, if ever, have to add numbers of different bases. However, it's all but guaranteed that you'll be asked to do it on the AP exam! The simplest way to do this is to convert each number to base 10, then add them. Unless specifically told otherwise, you can leave your answer in base 10.

01102 + A016 + 428 + 32

\_6\_ + \_160\_ + \_34\_ + \_32\_ = 232

Add the following numbers; give your answer in decimal (base 10).

01002 + 1A16 + 208 + 20 = \_\_\_\_\_

**We will revisit this topic throughout the year. It will be on the AP exam. If you don't understand, please talk to me during office hours.** If it takes you a long time to solve these types of problems, don't worry. If you understand the underlying rules, **you can now solve any problem in any base**. In this class, as well as on the AP exam, only binary, octal, and hexadecimal bases will be tested. But you should be aware that positional notation can be applied to any base.