**Binary Worksheet**

The purpose of this lab is to practice with some binary number representations. Binary numbers are the dough we use to represent all information inside the computer.

**Part I: Binary to decimal conversion**

As an example, let’s look at the binary number 11010.

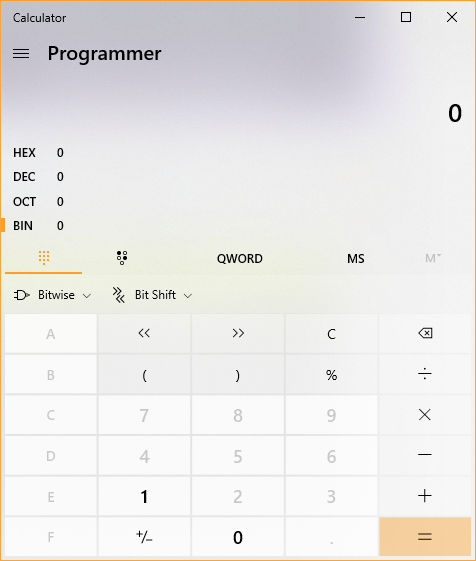
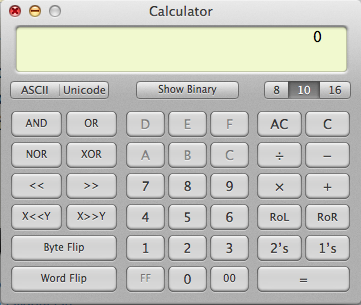
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **0** | **1** | **1** | **0** | **1** | **0** |
| 25 = 32 | 24 = 16 | 23 = 8 | 22 = 4 | 21 = 2 | 20 = 1 |

To determine its value in base 10, all we need to do is add up the powers of 2 wherever we see a “1” in our binary number. In this case we have 16 + 8 + 2 = 26.

**Practice: Convert each of the following binary numbers into decimal.**

|  |  |  |
| --- | --- | --- |
| ***Binary*** | ***Which powers of 2 are we adding?*** | ***Decimal answer*** |
| **101010** | 21+23+25 | 42 |
| **1110** | 21+22+23 | 14 |
| **1010** | 21+23 | 10 |
| **111110** | 21+22+23+24+25 | 62 |
| **1001100** | 22+23+26 | 76 |
| **101100** | 22+23+25 | 44 |

We can check our answers by using an online calculator, such as the one built in to Windows or MacOS.

**Part II: Decimal to binary conversion: the “Binary Store”**

To convert a number n into binary, pretend you have $n to spend at the binary store. Your strategy is to buy the most valuable gifts you can. The price list at the store is simply the powers of two: 1, 2, 4, 8, 16, 32, 64, etc.

Example: Let’s say you have $60 to start with.

First, buy a $32 item. This will leave you with 60 – 32 = 28.

Next, buy a $16 item. This leaves you with 28 – 16 = 12.

Next, buy an $8 item. Now you have 12 – 8 = 4.

Finally you can buy a $4 item, and all your money is spent.

Your cashier receipt shows that you bought items costing: 32 + 16 + 8 + 4. Each of these numbers is a power of 2, so we can fill in the blanks. “1” means you bought that item, and “0” means you didn’t. In this case, we did not buy any $2 or $1 item.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **1** | **1** | **1** | **1** | **0** | **0** |
| 25 = 32 | 24 = 16 | 23 = 8 | 22 = 4 | 21 = 2 | 20 = 1 |

Thus, the decimal number 60 is equivalent to the binary 111100.

**Try these examples. Convert decimal numbers to binary:**

|  |  |  |
| --- | --- | --- |
| ***Decimal*** | ***What can you buy at the binary store?*** | ***Binary answer*** |
| **86** | 64+16+4+2 | 1010110 |
| **18** | 16+2 | 10010 |
| **35** | 32+2+1 | 100011 |
| **50** | 32+16+2 | 110010 |
| **101** | 64+32+4+1 | 1100101 |
| **16** | 16 | 10000 |

**Observations**

Looking at the above examples, let’s see if we can detect some patterns.

1. Look at binary representations of the even numbers we did earlier, such as 86, 18 and 50. What do they all have in common?

None of them had a 20 in the binary answer.

1. Look at the odd numbers like 35 and 101. What do their binary representations have in common?

All of them had a 20 in the binary answer.

1. Let’s say you’ve just written down the binary form of some number x.  
   What will happen to the value of the number if we put a “0” at the beginning of x?   
     
   The number would remain the same.
2. What would happen if we put a “0” at the end?

The number would increase.

1. If a binary number has all 1s in it (e.g. 111, 1111, 11111), what kind of number is this? Yes, it would be odd, yes it will have all 1s – be more specific.

It would be an one bit.

**Part III: Binary shorthand**

Binary numbers can be painful to write or type when they get long. For example, when studying computer colors, it turns out that to completely specify a color, it takes 24 bits! (“bit” = binary digit) Here is an example: 100110010011001111111111

There are 2 kinds of shorthand to help us. They are called octal and hexadecimal.

***Octal***

Octal means base 8, and 8 is 23. Thus, one octal digit can stand for 3 bits.

To experiment with octal numbers, let’s play with the online calculator again. Set it up so that it will convert from base 8 to base 2.

**Ask it to do these conversions, and write down the answers:**

|  |  |
| --- | --- |
| ***Octal*** | ***Binary*** |
| **0005** | 0101 |
| **0011** | 1001 |
| **1020** | 0010 0001 000 |
| **0606** | 0001 1000 0110 |
| **0126** | 0101 0110 |

In the examples 1020 and 606, how many bits are in our answers? Can you see the individual octal digits? Both have 12 bits.

**Now let’s try a few more octal-to-binary cases:**

|  |  |
| --- | --- |
| ***Octal*** | ***Binary*** |
| **6** | 0110 |
| **1** | 0001 |
| **0100** | 0100 0000 |
| **0612** | 0001 1000 1010 |
| **0123** | 0101 0011 |

**How many bits does it take to represent the octal “612”? What about “123”? Why do you think there is a difference?**

612 takes 9 bits and 123 takes 7 bits. There’s a difference because they are different binary numbers.

Next, set up the online calculator so that it will go the other way: from base 2 to base 8. **Ask the calculator to do these conversions:**

|  |  |
| --- | --- |
| ***Binary*** | ***Octal*** |
| **110** | 6 |
| **110111** | 67 |
| **010111000** | 270 |

**Can you see how the octal short hands are determined? Explain how it is done, in your own words.**

Octal shorthand uses base 8. This means that it uses digits 0 through 7.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Hexadecimal***

The second type of shorthand is called hexadecimal, or just “hex” for short. This means base 16, and 16 = 24. Thus, each hex digit stands for 4 bits, making hex even more efficient than octal as a shorthand for binary.

Set up the online calculator to go from base 16 to base 2. Ask it to do these conversions:

|  |  |
| --- | --- |
| ***Hex*** | ***Binary*** |
| **9** | 1001 |
| **3** | 0011 |
| **93** | 1001 0011 |
| **903** | 1001 000 0011 |
| **309** | 0011 0000 1001 |

Again, note the number of bits in our answers.

Now, let’s go the other way. Let’s ask the online calculator to convert from binary to hexadecimal. Set up the calculator to convert from base 2 to base 16.

**Ask it to do these conversions:**

|  |  |
| --- | --- |
| ***Binary*** | ***Hex*** |
| **0101** | 5 |
| **0011** | 3 |
| **00110101** | 35 |
| **10010000** | 90 |
| **10011001** | 99 |
| **1100** | C |
| **10011100** | 9C |

Do the last two results look strange? It turns out that in base 16, we must have 16 different symbols for our digits. The digits 0-9 are already familiar to us, but now we need special symbols to represent digit values 10-15.

To fully understand what is going on with hex shorthand, we can ask the online calculator to convert the numbers 10 thru 15 from decimal to hex. Write the results here:

|  |  |
| --- | --- |
| ***Decimal*** | ***Hex*** |
| **9** | 9 |
| **10** | A |
| **11** | B |
| **12** | C |
| **13** | D |
| **14** | E |
| **15** | F |
| **16** | 10 |