CPSC 231: Asst 2 Extra Material: Number conversions and Non-decimal based math: Answers

Jordan Kidney

January 24, 2006

1 Common notation and useful link

- N_B = N is the number, B is the base the number is represented in.
- http://www.cstc.org/data/resources/60/convexp.html
- http://www.mathpath.org/concepts/Num/conv.htm
- http://www.math.grin.edu/~rebelsky/Courses/152/97F/Readings/student-binary.html (The Binary System)

2 Number conversions

One way to do number conversions is to repeatedly divide the decimal number by the base in which it is to be converted, until the quotient becomes zero. As the number is divided, the remainders - in reverse order - form the digits of the number in the other base. Faster ways to do conversions will be demonstrated , were possible, in the answers given below.

2.1 Decimal to binary

Convert the following from base 10 (decimal) to base 2 (binary):

2.1.1 6_{10}

1. 6/2 = 3 remainder 0
2. 3/2 = 1 remainder 1
3. 1/2 = 0 remainder 1

answer: 110_2

2.1.2 12_{10}

1. $12/2 = 6$ remainder 0
2. $6/2 = 3$ remainder 0
3. $3/2 = 1$ remainder 1
4. $1/2 = 0$ remainder 1

answer: 1100_2

2.1.3 23_{10}

1. $23/2 = 11$ remainder 1
2. $11/2 = 5$ remainder 1
3. $5/2 = 2$ remainder 1
4. $2/2 = 1$ remainder 0
5. $1/2 = 0$ remainder 1

answer: 10111_2

2.1.4 256_{10}

1. $256/2 = 128$ remainder 0
2. $128/2 = 64$ remainder 0
3. $64/2 = 32$ remainder 0
4. $34/2 = 16$ remainder 0
5. $16/2 = 8$ remainder 0
6. $8/2 = 4$ remainder 0
7. $4/2 = 2$ remainder 0
8. $2/2 = 1$ remainder 0
9. $1/2 = 0$ remainder 1

answer: 100000000_2

2.1.5 1529_{10}

1. $1529/2 = 764$ remainder 1
2. $764/2 = 382$ remainder 0
3. $382/2 = 191$ remainder 0
4. $191/2 = 95$ remainder 1
5. $95/2 = 47$ remainder 1
6. $47/2 = 23$ remainder 1
7. $23/2 = 11$ remainder 1
8. $11/2 = 5$ remainder 1
9. $5/2 = 2$ remainder 1
10. $2/2 = 1$ remainder 0
11. $1/2 = 0$ remainder 1

answer: 10111111001_2

2.2 Decimal to octal or hexadecimal or binary

Convert the following from base 10 (decimal) to base 2 (binary), base 8 (octal) and base 16 (hexadecimal) :

Note: to solve these questions you could just follow the same process done in the previous section. In the case of octal or hexadecimal you would just divide by 8 or 16 respectively. Now there is a faster way that both of these can be done by just doing one conversion. This comes from the fact that 8 and 16 are just powers of 2 ($8 = 2^3$,$16 = 2^4$). Now these number systems are based upon the position of each number. This means that for the base , that the number is represented in, there is a digit for each power of the base. For example, the number 146_{10} (decimal) can also be written as $146_{10} = (1 \times 10^2) + (4 \times 10^1) + (6 \times 10^0)$. Now you may be asking yourself how does this help me ? well this means in the end that one octal digit is equal to 3 digits in binary, and a single hexadecimal digit is equivalent to 4 digits in binary. To do the conversion from the binary to to octal or hexadecimal you just take the equivalent number of digits and use the following formula to convert it to an octal or hexadecimal digit.

- Binary number $= b_3b_2b_1$, octal digit $= (b_3 \times 2^2) + (b_2 \times 2^1) + (b_1 \times 2^0)$. For example $101_2 = (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) = (1 \times 4) + (0 \times 2) + (1 \times 1) = 5_{10} = 5_8$
- Binary number $= b_4b_3b_2b_1$, hexadecimal digit $= (b_4 \times 2^3) + (b_3 \times 2^2) + (b_2 \times 2^1) + (b_1 \times 2^0)$. For example $1101_2 = (1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) = (1 \times 8) + (1 \times 4) + (0 \times 2) + (1 \times 1) = 13_{10} = D_{16}$

To convert the entire binary number to octal or hexadecimal, start at the right hand side of the number and take the first 3 or r digits and convert them. This then becomes the first digit for the hex or octal number on the right hand side. Continue this process for the nest digits in the number. This process is shown in the answers given below to each question.

2.2.1 12_{10}

Binary number: $12_{10} = 1100_2$.
Octal number:

1. Take the first three bits and convert: $100_2 = 4_{10} = 4_8$
2. Now take the next three bits, (since we are at the front of the number just add leading zeros to make up the three bits we need). $001_2 = 1_{10} = 1_8$

so the octal number now becomes 14_8
Hex number:

1. Take the first four bits and convert: $1100_2 = 12_{10} = C_{16}$

so the hex number now becomes C_{16}

2.2.2 35_{10}

Binary number: $35_{10} = 100011_2$.
Octal number:

1. $011_2 = 3_{10} = 3_8$
2. $100_2 = 4_{10} = 4_8$

so the octal number now becomes 43_8
Hex number:

1. $0011_2 = 3_{10} = 3_{16}$
2. $0010_2 = 2_{10} = 2_{16}$

so the hex number now becomes 23_{16}

2.2.3 256_{10}

Binary number: $256_{10} = 100000000_2$.
Octal number:

1. $000_2 = 0_{10} = 0_8$
2. $000_2 = 0_{10} = 0_8$

3. $100_2 = 4_{10} = 4_8$

so the octal number now becomes 400_8
Hex number:

1. $0000_2 = 0_{10} = 0_{16}$
2. $0000_2 = 0_{10} = 0_{16}$
3. $0001_2 = 1_{10} = 1_{16}$

so the hex number now becomes 100_{16}

2.2.4 512_{10}

Binary number: $512_{10} = 1000000000_2$.
Octal number:

1. $000_2 = 0_{10} = 0_8$
2. $000_2 = 0_{10} = 0_8$
3. $000_2 = 0_{10} = 0_8$
4. $001_2 = 1_{10} = 1_8$

so the octal number now becomes 1000_8
Hex number:

1. $0000_2 = 0_{10} = 0_{16}$
2. $0000_2 = 0_{10} = 0_{16}$
3. $0010_2 = 2_{10} = 2_{16}$

so the hex number now becomes 200_{16}

2.2.5 1189_{10}

Binary number: $1189_{10} = 10010100101_2$.
Octal number:

1. $101_2 = 5_{10} = 5_8$
2. $100_2 = 4_{10} = 4_8$
3. $010_2 = 2_{10} = 2_8$
4. $010_2 = 2_{10} = 2_8$

so the octal number now becomes 2245_8
Hex number:

1. $0101_2 = 5_{10} = 5_{16}$
2. $1010_2 = 10_{10} = A_{16}$
3. $0100_2 = 4_{10} = 4_{16}$

so the hex number now becomes $4A5_{16}$

3 Non-decimal based math

Perform the following binary additions and subtractions:

Note: For the subtractions use borrows (where needed) rather than employing the complement and add technique.

3.0.6 $10010101_2 + 01101001_2$

Carry							1	
	1	0	0	1	0	1	0	1
+	0	1	1	0	1	0	0	1
Answer	1	1	1	1	1	1	1	1

3.0.7 $1110_2 + 0010_2$

Carry		1	1		
		1	1	1	0
+		0	0	1	0
Answer	1	0	0	0	0

3.0.8 $01100110_2 - 01100001_2$

Borrow							1	
	0	1	1	0	0	1	0	0
-	0	1	1	0	0	0	0	1
Answer	0	0	0	0	0	1	0	1

3.0.9 $0110_2 - 0010_2$

Borrow				
	0	1	1	0
-	0	0	1	0
Answer	0	1	0	0