Notes for Lecture 6

Representations of integers in different bases

We are commonly using the **decimal system** of writing numbers. For instance:

$$1234 = 1 \cdot 10^3 + 2 \cdot 10^2 + 3 \cdot 10^1 + 4 \cdot 10^0.$$

10 is called the base, and $1, 2, 3, 4$ are the digits in base 10. To emphasize that we are using base 10, we will write $1234 = (1234)_{10}$. Likewise, we write

$$(1234)_b = 1 \cdot b^3 + 2 \cdot b^2 + 3 \cdot b^1 + 4 \cdot b^0.$$

In this example, $b > 4$, because, if b is the base, then the digits have to be in $\{0, 1, \ldots, b - 1\}$.

Comment. In the above examples, it is somewhat ambiguous to say whether 1 or 4 is the first or last digit. To avoid confusion, one refers to 4 as the **least significant digit** and 1 as the **most significant digit**.

Example 33. $25 = 16 + 8 + 1 = \boxed{1} \cdot 2^4 + \boxed{1} \cdot 2^3 + \boxed{0} \cdot 2^2 + \boxed{0} \cdot 2^1 + \boxed{1} \cdot 2^0$.
Accordingly, $25 = (11001)_2$.

While the approach of the previous example works well for small examples when working by hand (if we are comfortable with powers of 2), the next example illustrates a more algorithmic approach.

Example 34. Express 49 in base 2.

Solution.

- $49 = 24 \cdot 2 + \boxed{1}$. Hence, $49 = (\ldots 1)_2$ where $\ldots$ are the digits for 24.
- $24 = 12 \cdot 2 + \boxed{0}$. Hence, $49 = (\ldots 01)_2$ where $\ldots$ are the digits for 12.
- $12 = 6 \cdot 2 + \boxed{0}$. Hence, $49 = (\ldots 001)_2$ where $\ldots$ are the digits for 6.
- $6 = 3 \cdot 2 + \boxed{0}$. Hence, $49 = (\ldots 0001)_2$ where $\ldots$ are the digits for 3.
- $3 = 1 \cdot 2 + \boxed{1}$. Hence, $49 = (\ldots 10001)_2$ where $\ldots$ are the digits for 1.
- $1 = 0 \cdot 2 + \boxed{1}$. Hence, $49 = (110001)_2$.

Other bases.
What is 49 in base 3? $49 = 16 \cdot 3 + \boxed{1}$, $16 = 5 \cdot 3 + \boxed{1}$, $5 = 1 \cdot 3 + \boxed{2}$, $1 = 0 \cdot 3 + \boxed{1}$. Hence, $49 = (1211)_3$.
What is 49 in base 5? $49 = (144)_5$.
What is 49 in base 7? $49 = (100)_7$.

Example 35. Bases 2, 8 and 16 (binary, octal and hexadecimal) are commonly used in computer applications.

For instance, in JavaScript or Python, $0\mathrm{b}\ldots$ means $(\ldots)_2$, $0\mathrm{o}\ldots$ means $(\ldots)_8$, and $0\mathrm{x}\ldots$ means $(\ldots)_{16}$.
The digits $0, 1, \ldots, 15$ in hexadecimal are typically written as $0, 1, \ldots, 9, A, B, C, D, E, F$.
Example. FACE value in decimal? $(FACE)_{16} = 15 \cdot 16^3 + 10 \cdot 16^2 + 12 \cdot 16 + 14 = 64206$
Practical example. `chmod 664 file.tex` (change file permission)

664 are octal digits, consisting of three bits: $1 = (001)_2$ execute (x), $2 = (010)_2$ write (w), $4 = (100)_2$ read (r)
Hence, 664 means rw,rw,r. What is rwx,rx,-? 750
By the way, a fourth (leading) digit can be specified (setting the flags: setuid, setgid, and sticky).

Armin Straub
straub@southalabama.edu

Modern ciphers

Example 36. For modern ciphers, we will change the alphabet from $A, B, ..., Z$ to $0, 1$. One of the most common ways of encoding text is **ASCII**.

In ASCII (American Standard Code for Information Interchange), each letter is represented using 8 bits (1 byte). Among the $2^8 = 256$ many characters are the usual letters, as well as common symbols.

For instance: $\text{space} = (20)_{16}$, $\text{“0”} = (30)_{16}$, $A = (41)_{16} = (0100, 0001)_2 = 65$, $a = (61)_{16} = (0110, 0001)_2 = 97$

See, for instance, `http://www.asciitable.com` for the full table.

Example 37. The new (8/2018) insignia of **FinCEN** features binary digits. What do they mean?

01000110 01101001 01101110 01000011 01000101 01001110 `https://www.fincen.gov`

By the way. If you ever have more than $\$10,000$ in foreign accounts, you must file a report to FinCEN.

One-time pad

Definition 38. The “exclusive or” (XOR), often written $\oplus$, is defined bitwise:

	0	0	1	1
$\oplus$	0	1	0	1
$=$	0	1	1	0

Note. On the level of individual bits, this is just addition modulo 2.

By the way. Best thing about a boolean: even if you are wrong, you are only off by a bit.

Example 39. $1011 \oplus 1111 = 0100$

Example 40. Observe that $a \oplus b \oplus b = a$.

One way to see that is to think bitwise in terms of addition modulo 2. Then, $a + b + b = a + 2b \equiv a \pmod{2}$.

A **one-time pad** works as follows. We use a key k of the same length as the message m. Then the ciphertext is

$$c = E_k(m) = m \oplus k.$$

To decipher, we use $m = D_k(c) = c \oplus k$.

As the name indicates, we must never use this key again!

Note. Observe that encryption and decryption are the same routine.

Comment. If that is helpful, a one-time pad is doing exactly the same as the Vigenere cipher if we use a key of the same length as the message (also, we use $0, 1$ as our letters instead of the classical $A, B, ..., Z$).

Example 41. Using a one-time pad with key $k = 1100, 0011$, what is the message $m = 1010, 1010$ encrypted to?

Solution. $c = m \oplus k = 0110, 1001$

If a one-time pad (with perfectly random key) is used exactly once to encrypt a message, then **perfect confidentiality** is achieved (eavesdropping is hopeless).

Meaning that Eve intercepting the ciphertext can draw absolutely no conclusions about the plaintext (because, without information on the key, every text of the right length is actually possible and equally likely), see next example.