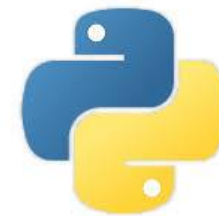


CMPT 120: Introduction to Computing Science and Programming 1

Binary Encoding / Representation



python™

Copyright © 2018, Liaqat Ali. Based on [CMPT 120 Study Guide](#) and [Think Python - How to Think Like a Computer Scientist](#), mainly. Some content may have been adapted from earlier course offerings by Diana Cukierman, Anne Lavergn, and Angelica Lim. Copyrights © to respective instructors. Icons copyright © to their respective owners.

Reminders

Liaqat Ali, Summer 2018.

One-Stop Access To Course Information

- **Course website**: One-stop access to all course information.

<http://www2.cs.sfu.ca/CourseCentral/120/liaqata/WebSite/index.html>

- Course Outline
- Exam Schedule
- Python Info
- **CourSys/Canvas** link
- Learning Outcomes
- Office Hours
- Textbook links
- and more...
- Grading Scheme
- Lab/Tutorial Info
- Assignments

- **Canvas**: Discussions forum - <https://canvas.sfu.ca/courses/39187>

- **CourSys**: Assignments submission, grades - www.coursys.sfu.ca

How to Learn in This Course?



- A** **Attend** Lectures & Labs
- R** **Read** / review Textbook/Slides/Notes
- R** **Reflect** and ask Questions
- O** **Organize** – your learning activities on weekly basis,
and finally...
- W** **Write** Code, **Write Code**, and **Write Code**.

Deliverables

1. Deliverables are due by the given date and time.
2. For the course, we are using IDLE to write and run our Python code.
3. You can use the CSIL lab computers outside your lab hours.
4. Plan ahead your assignments and other deliverables. Computer crash, network problems etc. are not acceptable excuses for delays in deliverables.
5. You may use online Python interpreters for running and testing your codes, such as:

<https://repl.it/languages/Python3>

Labs

1. Each lab has an assigned TA.
2. Attend your assigned lab and show your work to your TA for the participation marks.
3. Class enrolments and lab swaps are closed now.

Course Topics

1. General introduction
2. Algorithms, flow charts and pseudocode
3. Procedural programming in Python
4. Data types and Control Structures
5. **Fundamental algorithms**
6. **Binary encodings**
7. **Basics of computability and complexity**
8. **Basics of Recursion**
9. **Subject to time availability:**
 - **Basics of Data File management**

Today's Topics

1. Data Representation (Binary Encoding)

- ASCII
- Unicode
- Storage Units
- Binary Addition

1

Data Representation

Number Systems: Representation and Position Values

- In our common day life we use the Decimal Number System, only.
- We write **11**:
 - it is understood that we talking about the decimal number system.
 - We know it represents **Eleven**.
- But, what it **11** is a binary number?
 - It then would mean **3**.
- How can we can distingue number systems?
- One way is t use notation: **Number**_{base}
- So, write decimal **11** (base 10) as: **11**₁₀
 - **11**₁₀ thus represents **Eleven**.
 - Base 10 position values: **10²** **10¹** **10⁰**
- We write 11 in base 2 as: **11**₂
 - **11**₂ in Binary thus represents **three**.
 - Base 2 position values: **2²** **2¹** **2⁰**
- Similarly, we can write 11 in base 16 (hexadecimal) as: **11**₁₆
 - **11**₁₆ in hexadecimal means **Seventeen**.
 - Base 16 position values: **16²** **16¹** **16⁰**

Storage Units

- **Bit:** storage to represent a **binary** 0 or 1.
- **Byte:** a group of 8-bits.
- More bigger storage units (with approximation, as shown in Study Guide):

| Prefix | Symbol | Factor |
|-------------|--------|--|
| (no prefix) | | $2^0 = 1$ |
| kilo- | k | $2^{10} = 1024 \approx 10^3$ |
| mega- | M | $2^{20} = 1048576 \approx 10^6$ |
| giga- | G | $2^{30} = 1073741824 \approx 10^9$ |
| tera- | T | $2^{40} = 1099511627776 \approx 10^{12}$ |

- Example, “**12 megabytes**” is: 12×2^{20} bytes
- = **12,582,912 bytes** =>
12582912 × 8 bits = 100,663,296 bits .

- **More specifically:**

| Multiples of bytes | | | | | | V · T · E |
|--------------------|--------------|--|-------------------|--------------|-------------|-----------|
| Decimal | | | Binary | | | |
| Value | Metric | | Value | IEC | JEDEC | |
| 1000 | kB kilobyte | | 1024 | KiB kibibyte | KB kilobyte | |
| 1000 ² | MB megabyte | | 1024 ² | MiB mebibyte | MB megabyte | |
| 1000 ³ | GB gigabyte | | 1024 ³ | GiB gibibyte | GB gigabyte | |
| 1000 ⁴ | TB terabyte | | 1024 ⁴ | TiB tebibyte | – | |
| 1000 ⁵ | PB petabyte | | 1024 ⁵ | PiB pebibyte | – | |
| 1000 ⁶ | EB exabyte | | 1024 ⁶ | EiB exbibyte | – | |
| 1000 ⁷ | ZB zettabyte | | 1024 ⁷ | ZiB zebibyte | – | |
| 1000 ⁸ | YB yottabyte | | 1024 ⁸ | YiB yobibyte | – | |

Orders of magnitude of data

Quiz 1



Questions?