

Lab1_ID0356

Thursday, January 21, 2021 1:37 PM

62010256 อดิษฐ์ มนต์อักษร

62010966 นายสุทธินันท์ ญูใจ

ภาคผนวก A

การทดลองที่ 1 ข้อมูลและคณิตศาสตร์ในคอมพิวเตอร์

การทดลองนี้เป็นการทบทวนความเข้าใจและแบบฝึกหัดเสริมของเนื้อหาในบทที่ 2 เนื่องจากจำนวนบิตข้อมูลที่ยาวขึ้นจำเป็นต้องใช้โปรแกรมคอมพิวเตอร์ช่วยคำนวณแทน โดยมีวัตถุประสงค์ ดังต่อไปนี้

- เพื่อให้เข้าใจการแปลงและคณิตศาสตร์สำหรับเลขจำนวนเต็มฐานสองชนิดไม่มีเครื่องหมายและมีเครื่องหมายแบบ 2-Complement
- เพื่อให้เข้าใจการแปลงและคณิตศาสตร์สำหรับเลขทศนิยมฐานสองมาตรฐาน IEEE754 ชนิด Single Precision
- เพื่อให้เข้าใจรหัส ASCII และ Unicode สำหรับข้อมูลตัวอักษร

นอกจากเนื้อหาในบทที่ 2 แล้ว ผู้อ่านสามารถศึกษาเว็บเพจเพิ่มเติม เพื่อทำความเข้าใจอย่างลึกซึ้ง ได้แก่

- https://www.tutorialspoint.com/cprogramming/c_data_types.htm
- <https://www.ntu.edu.sg/home/ehchua/programming/java/datarepresentation.html>

ผู้อ่านจะพบว่าเนื้อหาในเว็บของมหาวิทยาลัยนันทราช ประเทศสิงคโปร์ เป็นการสอนพื้นฐานภาษา Java ใช้งานข้อมูลเป็นเลขฐานสองเหมือนกับภาษา C/C++ ในเว็บที่สอง การทดลองจะครอบคลุมเนื้อหาตามทฤษฎี โดยจะเริ่มจากเลขจำนวนเต็ม เลขทศนิยม และตัวอักษรตามลำดับ

A.1 การแปลงและคณิตศาสตร์สำหรับเลขฐานสองจำนวนเต็ม

A.1.1 การทดลอง

เนื่องจากการแปลงเลขฐานสิบเป็นฐานสองชนิดไม่มีเครื่องหมาย (unsigned) ผู้อ่านสามารถใช้เครื่องคิดเลขทางวิทยาศาสตร์ทั่วไป ดังนั้น การทดลองนี้จะเน้นที่การแปลงเป็นเลขฐานสองชนิดมีเครื่องหมายแบบ 2 Complement สอดคล้องกับเนื้อหาในหัวข้อที่ 2.2 โดยผ่านเว็บเบราว์เซอร์ที่ผู้อ่านกด กรอกหรือคลิกที่ ชื่อลิงค์ต่อไปนี้ http://www.free-test-online.com/binary/signed_converter.html ขอให้ผู้อ่านปฏิบัติตามการทดลอง ดังนี้

1. คลิกเลือกที่ปุ่ม Signed แล้วจึงกรอกเลข -123 ลงในกล่องข้อความ ดังรูปที่ A.1

รูปที่ A.1: กรอกเลข -123 ลงในกล่องข้อความ และคลิกเลือกที่ปุ่ม Signed เพื่อให้โปรแกรมแปลงเลขจำนวนเต็ม -123 เป็นเลขฐานสองชนิด Signed ชนิด 2 Complement

หน้าต่างมีลักษณะคล้ายเครื่องคิดเลข ประกอบด้วยปุ่มต่างๆ ดังนี้

- 'Bin2Dec' 'Dec2Bin' สำหรับแปลงเลขฐานสองเป็นฐานสิบไปและกลับ
- 'Dec2Hex' 'Hex2Dec' สำหรับแปลงเลขฐานสิบเป็นฐานสิบหกไปและกลับ
- 'Hex2Bin' 'Bin2Hex' สำหรับแปลงเลขฐานสองเป็นฐานสิบหกไปและกลับ
- ปุ่ม 0-9 และ A-F สำหรับกรอกตัวเลขฐานสิบและฐานสิบหก
- CL (Clear) สำหรับล้างค่าในกล่องข้อความให้เป็น 0
- RoR (Rotate Right) และ RoL (Rotate Left) สำหรับเลื่อนตัวเลขที่อยู่ในกล่องข้อความทางขวาและซ้าย ตามลำดับ
- ShR (Shift Right) และ ShL (Shift Left) โดยป้อนเลข 0 เข้ามาแทน
- 2's C (omplement) สำหรับแปลงเลขฐานสองให้เป็นค่า 2's Complement

- +/- สำหรับกลับเครื่องหมายของตัวเลขฐานสิบในกล่องข้อความ

2. กดปุ่มเครื่องหมาย 'Dec2Bin' เพื่อให้เป็นเลขฐานสองชนิด Signed ดังรูปที่ A.2

รูปที่ A.2: ผลลัพธ์จากการแปลงเลข -123 ให้เป็นเลขฐานสองชนิด Signed 2-Complement ความยาว 24 บิต

3. กดปุ่มเครื่องหมาย 'Bin2Hex' เพื่อแปลงเลขฐานสองที่ได้ให้เป็นเลขฐานสิบหกชนิด Signed ตามรูปที่ A.3

รูปที่ A.3: ผลลัพธ์จากการแปลงเลข -123 ให้เป็นเลขฐานสิบหก 6 หลักจากเลขฐานสองชนิด Signed 2-Complement ความยาว 24 บิต

4. กดปุ่ม Hex2Bin เพื่อแปลงผลลัพธ์เลขฐานสิบหกที่ได้กลับไปเป็นฐานสอง แล้วเลือกตัวเลขฐานสองทั้งหมด แล้วทำการคัดลอก (Copy) หรือกดปุ่ม Ctrl-C พร้อมกัน

5. คลิกบนชื่อลิงค์ต่อไปนี้ เพื่อเปิดหน้าเว็บสำหรับ บวก/ลบ/คูณ/หาร เลขจำนวนเต็ม ทั้งชนิด Unsigned และ Signed ต่อไปนี้

http://www.free-test-online.com/binary/binary_calculator.html

6. กดเลือกปุ่มออฟชั่น Signed ก่อนแล้วจึงทำการวาง (Paste) ลงในกล่องข้อความ เพื่อเปลี่ยนการทำงานให้อยู่ในโหมดตัวเลขฐานสองชนิดมีเครื่องหมายตามรูปที่ A.4

รูปที่ A.4: หน้าต่างวางเลขการแปลงเลข -123 ให้เป็นเลขฐานสองชนิด Signed 2-Complement ความยาว 24 บิต

7. กดปุ่ม '-' เพื่อทำการกระบวนการลบเลข แล้ววาง (Paste) เลข -123 อีกรอบในกล่องข้อความที่ว่างลง
8. กดปุ่ม = เพื่อแสดงผลลัพธ์

รูปที่ A.5: ผลลัพธ์เลขการแปลงเลข (-123) - (-123) ให้เป็นเลขฐานสองชนิด Signed 2-Complement ความยาว 24 บิต

ในรูปที่ A.5 แสดงให้เห็นว่า $-123 - (-123) = 0$

A.1.2 กิจกรรมท้ายการทดลอง

จงทำการทดลองและตอบคำถามต่อไปนี้ โดยแสดงวิธีทำตามเนื้อหาในหัวข้อที่ 2.2.2 และตรวจคำตอบตามวิธีทำการทดลองที่ได้ทำไป

1. จงแปลงเลขฐานสิบชนิดไม่มีเครื่องหมายต่อไปนี้ให้เป็นเลขฐานสอง 24 บิตและฐานสิบหกจำนวน 6 หลัก และบันทึกผลลัพธ์ที่ได้ลงในตาราง

ฐานสิบ	ฐานสอง	ฐานสิบหก
7	00000111	7
8	00001000	8
15	00001111	F
16	00010000	10
255	11111111	FF
256	100000000	100
65535	1111111111111111	FFFF
65536	1000000000000000	10000

2. จงแปลงเลขฐานสิบต่อไปนี้ให้เป็นเลขฐานสองและฐานสิบหกชนิดมีเครื่องหมายแบบ 2-Complement และบันทึกผลลัพธ์ที่ได้ลงในตาราง

ฐานสิบ	ฐานสอง	ฐานสิบหก
+1	00000001	1
-1	11111111	FF
+15	00001111	F
-16	111111110000	F0
+255	0000000011111111	FF
-256	1111111110000000	F0
+65535	000000001111111111111111	FFFF
-65536	111111110000000000000000	F0000

3. จงบวกเลข 2-Complement ต่อไปนี้ แล้วบันทึกผลลัพธ์เป็นฐานสอง ฐานสิบ ข้อผิดพลาดที่แจ้งเตือนและอธิบายเหตุผลว่าทำไมจึงไม่ตรงกัน

- $10000000000000000000000000000000 + 00000000000000000000000000000001$
 - ผลลัพธ์ฐานสอง = 10000000000000000000000000000001
 - ผลลัพธ์ฐานสิบ = -2147483647 10 ฐานสิบหก FFFFFFFF 16
 - ข้อผิดพลาดที่แจ้งเตือน.....
 - เหตุผล.....
- $10000000000000000000000000000000 + 10000000000000000000000000000000$
 - ผลลัพธ์ฐานสอง = 10000000000000000000000000000000
 - ผลลัพธ์ฐานสิบ = 0 10 ฐานสิบหก 00000000 16
 - ข้อผิดพลาดที่แจ้งเตือน...Overflow.....

[illegible]

- ผลลัพธ์ฐานสิบ = -2147483649 10 จำนวนลบ 3 16

- เหตุผล not enough binary digits to display the number.

[illegible]

- ผลลัพธ์ฐานสิบ = 10 งานสิบตก 16

- เหตุผล.....

A.2 การแปลงและคณิตศาสตร์สำหรับมาตรฐาน IEEE754

การทดลองเพื่อให้เข้าใจการแปลงเลขจำนวนจริงฐานสิบให้เป็นเลขฐานสองตามรูปแบบและฝึกการคำนวณ โดยใช้คณิตศาสตร์มาตรฐาน IEEE754 Single Precision มีความสอดคล้องกับเนื้อหาในหัวข้อที่ 2.6

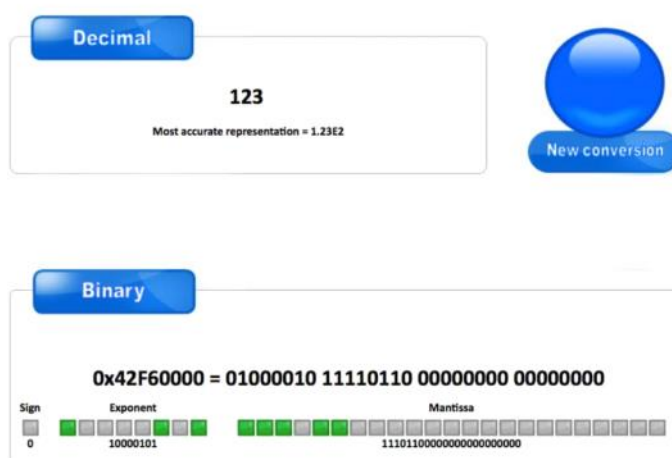
A.2.1 การทดลองสำหรับ Single-Precision

การทดลองนี้จะเน้นที่การแปลงเลขจำนวนจริงให้เป็นเลขฐานสองทศนิยมชนิดลอยตัว สอดคล้องกับเนื้อหาในหัวข้อที่ 2.6 ในรูปแบบ Single Precision โดยผ่านเว็บเบราว์เซอร์ที่ผู้อ่านถนัด กรอกหรือคลิกที่ชื่อลิงค์ต่อไปนี้

http://www.binaryconvert.com/convert_float.html

เมื่อเว็บเพจปรากฏขึ้น ขอให้ผู้อ่านปฏิบัติตามการทดลอง ดังนี้

1. กรอกเลข 123 ลงในกล่องข้อความ แล้วกดปุ่ม Convert to binary ได้รูปที่ A.6



รูปที่ A.6: ผลลัพธ์จากการแปลงเลข 123.0 ให้เป็นเลขฐานสองชนิด Single Precision

การเรียงตัวของผลลัพธ์เลขฐานสิบหกทางซ้ายมือมาจากเลขฐานสองทางขวามือ ซึ่งเกิดจากบิตข้อมูลทั้งหมด 32 บิตตามรูปแบบของมาตรฐาน IEEE754 ชนิด Single Precision โปรดสังเกต กล่องสี่เหลี่ยมสีเขียวตรงกับบิตที่เป็น '1' กล่องสีเทาตรงกับบิตที่เป็น '0' 0x หมายถึง เลขฐานสิบหก

2. กรอกเลข -123.0 ลงในกล่องข้อความ แล้วกดปุ่ม Convert to binary ได้รูปที่ A.7



รูปที่ A.7: ผลลัพธ์จากการแปลงเลข -123.0 ให้เป็นเลขฐานสองตามมาตรฐาน IEEE754 ชนิด Single Precision

โปรดสังเกตตำแหน่งของกล่องสี่เหลี่ยมหรือสี่เหลี่ยมที่ตรงกับบิต Sign Exponent และ Mantissa ดังนั้น เราจะเห็นว่าเฉพาะ Sign ที่มีการเปลี่ยนแปลง

3. คลิกบนลิงค์นี้ เพื่อทดลองบวกและคูณเลขในรูปแบบ Single Precision ด้วยลิงค์ต่อไปนี้ <http://weitz.de/ieee/> เลื่อนหน้าเว็บลงไปที่ด้านล่างสุด เพื่อค้นหาแถบเมนูตามรูปที่ A.8

binary16 binary32 binary64 binary128

รูปที่ A.8: เมนูด้านล่างสุดของหน้าเว็บ เพื่อเลือกเลขฐานสองชนิด Single Precision (Binary32) และ Double Precision (Binary64)

4. เลื่อนหน้าเว็บกลับไปด้านบนสุดเพื่อกรอกเลข -123.0 ลงในกล่องข้อความซ้ายบน และ กรอกเลข 123.0 ลงในกล่องข้อความถัดลงมา แล้วกดปุ่ม + แล้วจะได้ผลลัพธ์ดังรูปต่อไปนี้

IEEE 754 Calculator
(See info at bottom of page.)

	Sign	Significand	Exponent
-123.0	1 -	1 .111011000000000000000000 1.921875 0xC2F60000 0b11000010111101100000000000000000	10000101 +6
123.0	0 +	1 .111011000000000000000000 1.921875 0x42F60000 0b01000010111101100000000000000000	10000101 +6
<div style="display: flex; justify-content: space-around; margin-bottom: 10px;"> + - × / </div>			
0.0	0 +	0 .000000000000000000000000 0.0 0x00000000 0b00000000000000000000000000000000	00000000 +0

รูปที่ A.9: ผลลัพธ์จากการบวกเลข -123.0+123.0 ให้เป็นเลขฐานสองชนิด Single Precision

จะสังเกตเห็นว่า ผลลัพธ์ที่ได้เรียกว่า True Zero ตามตารางที่ 2.12

5. กดปุ่ม \times (คูณ) แล้วจะได้ผลลัพธ์ของ -123×123 ดังรูปต่อไปนี้

IEEE 754 Calculator
(See info at bottom of page.)

	Sign	Significand	Exponent
-123.0	1 -	1 .111011000000000000000000 1.921875 0xC2F60000 0b11000010111101100000000000000000	10000101 +6
123.0	0 +	1 .111011000000000000000000 1.921875 0x42F60000 0b01000010111101100000000000000000	10000101 +6
<div style="display: flex; justify-content: space-around; margin-bottom: 10px;"> + - × / </div>			
-15129.0	1 -	1 .110110001100100000000000 1.8468018 0xC66C6400 0b11000110011011000110010000000000	10001100 +13

รูปที่ A.10: ผลลัพธ์จากการคูณเลข -123.0×123.0 ให้เป็นเลขฐานสองชนิด Single Precision

A.2.2 กิจกรรมท้ายการทดลอง

จงใช้เว็บเพจลิงค์ต่อไปนี้ในการตอบคำถาม

<https://www.h-schmidt.net/FloatConverter/IEEE754.html>

Tools & Thoughts

IEEE-754 Floating Point Converter

Translations: [de](#)

This page allows you to convert between the decimal representation of numbers (like "1.02") and the binary format used by all modern CPUs (IEEE 754 floating point).

IEEE 754 Converter (JavaScript), V0.22

	Sign	Exponent	Mantissa
Value:	+1	2^{-126} (denormalized)	0.0 (denormalized)
Encoded as:	0	0	0
Binary:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

You entered: 0

Value actually stored in float: 0

Error due to conversion: 0

Binary Representation: 00000000000000000000000000000000

Hexadecimal Representation: 0x00000000

รูปที่ A.11: เว็บไซต์สำหรับการตอบคำถามเพื่อสร้างเลขหรือแปลงเลขฐานสิบด้วยมาตรฐาน IEEE754 Single Precision การกดเลือกคือทำให้บิตนั้นเท่ากับ '1'

โดยแสดงวิธีทำตามเนื้อหาในหัวข้อที่ 2.6 และตรวจคำตอบตามวิธีทำการทดลองที่ได้ทำไป และกรอกผลลัพธ์ลงบนเส้นประที่จัดไว้ให้เท่านั้น ยกตัวอย่างเช่น

1. จงสร้างเลข -0.0_{10} โดยการกดเลือกปุ่มสี่เหลี่ยมในส่วน Sign เท่านั้น
 เลขฐานสอง = 1 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0
 ฐานสิบหก = 8 0 0 0 0 0 0 0
2. จงสร้างเลข -1.0_{10} โดยการกดเลือกปุ่มสี่เหลี่ยมในส่วน Exponent เท่านั้น ต่อจากข้อที่แล้ว
 เลขฐานสอง = 1 0 1 1 | 1 1 1 1 | 1 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0
 ฐานสิบหก = B F 8 0 0 0 0 0
3. จงสร้างเลข -1.5_{10} โดยการกดเลือกปุ่มสี่เหลี่ยมในส่วน Mantissa เท่านั้น ต่อจากข้อที่แล้ว
 เลขฐานสอง = 1 0 1 1 | 1 1 1 1 | 1 0 1 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0
 ฐานสิบหก = B F C 0 0 0 0 0
4. จงสร้างเลข $1.17549435082 \times 10^{-38}$ ซึ่งเป็นค่านอมนัลไลซ์ที่น้อยที่สุด (Normalized)
 เลขฐานสอง = 0 0 0 0 | 0 0 0 0 | 1 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0
 ฐานสิบหก = 0 0 8 0 0 0 0 0

A.3 รหัสของข้อมูลตัวอักษร

A.3.1 การทดลอง

การทดลองในหัวข้อนี้จะเป็นการแปลงรหัสตัวอักษรภาษาอังกฤษและไทย เป็นรหัส ASCII และ Unicode ชนิด UCS-2 ตามเนื้อหาในหัวข้อ 2.7 ผ่านทางเว็บไซต์ <https://www.branah.com/ascii-converter> ที่มีนักพัฒนาเพื่อเผยแพร่ความรู้เป็นวิทยาทานเช่นเดียวกับเว็บที่ได้ทดลองมา

1. เปิดเว็บตามลิงค์ต่อไปนี้ หรือ กดปุ่มซ้ายบนชื่อลิงค์
<https://www.branah.com/ascii-converter>
2. กรอกข้อความต่อไปนี้ ลงในกล่องข้อความ ASCII
ไ ท ย ก ข ค a b c
โปรดสังเกต ระหว่างตัวอักษรมี ช่องว่าง 1 ตัวอักษรเสมอ
3. กดปุ่ม Convert ซ้ายบนสุด จะได้ผลลัพธ์ดังรูปต่อไปนี้

ASCII Converter - Hex, decimal, binary, base64, and ASCII converter

The screenshot shows a web interface for an ASCII converter. It has five sections, each with a 'Convert' button and a text input field. The first section is for ASCII, showing the input 'ไ ท ย ก ข ค a b c' and the output 'e44 e17 e22 e01 e02 e04 61 62 63'. The second section is for Hex, showing the input 'ไ ท ย ก ข ค a b c' and the output 'e44 e17 e22 e01 e02 e04 61 62 63'. The third section is for Decimal, showing the input 'ไ ท ย ก ข ค a b c' and the output '3652 3607 3618 3585 3586 3588 097 098 099'. The fourth section is for Binary, showing the input 'ไ ท ย ก ข ค a b c' and the output '111001000100 111000010111 111000100010 111000000001 111000000010 111000000100 01100001 01100010 01100011'. The fifth section is for Base64, showing the input 'ไ ท ย ก ข ค a b c' and the output 'RCAXIClgASACIAQgYSBIIGM='.

รูปที่ A.12: ผลลัพธ์จากการกรอกและแปลงตัวอักษร ไ ท ย ก ข ค a b c เป็นรหัสต่างๆ

4. กล่องข้อความ Hex จะแสดงค่า Unicode สำหรับภาษาไทย และ ASCII สำหรับภาษาอังกฤษ ในรูปผู้เขียนได้กดเลือก Remove 0x เพื่อความสะดวกในการอ่านค่า

A.3.2 กิจกรรมท้ายการทดลอง

1. จงอธิบายวิธีการหาค่าฐานสิบ 0 - 9 จากระหัส ASCII ของตัวอักษร 0 - 9
2. จงอธิบายวิธีการหาค่าฐานสิบ 0 - 9 จากระหัส Unicode ของตัวอักษร ๐ - ๙
3. จงเปิดเว็บที่มีข้อความภาษาไทย เช่น เว็บข่าว แล้วทดลองเปลี่ยนการนำเสนอบนจอเพื่อ View source เช่น Google Chrome ใช้เมนู Tool-> View Source แล้ว Find หรือกดปุ่ม CTRL-F คำว่า charset ว่ามีค่าเท่ากับ utf-8 หรือไม่ เพราะเหตุใด

① จากตาราง ASCII จะได้ว่า '0' = 48 '1' = 49 '2' = 50 ... '9' = 57
ดังนั้นการแปลงค่าฐานสิบของตัวอักษร 0-9 ทำได้โดยนำรหัส ASCII และลบด้วยค่า 48 #

② จากรหัส Unicode จะได้ว่า '0' = 3664 '๑' = 3665 '๒' = 3666 ... '๙' = 3673
ดังนั้นการแปลงค่าฐานสิบของตัวอักษร ๐-๙ ทำได้โดยนำรหัส Unicode และลบด้วยค่า 3664 #

③ เป็น UTF-8 เพราะว่าตัวอักษรในภาษาไทยส่วนใหญ่เป็นอักษร 1 ไบต์ โดยที่ตัวอักษร 1 ไบต์ UTF-8 อักษรจะเก็บด้วย ความยาว 1-4 ไบต์ โดยตัวอักษรในตาราง ASCII จะเก็บด้วยความยาว 1 ไบต์ ส่วนตัวอักษรภาษาอื่น ๆ จะเก็บด้วยความยาวมากกว่า 1 ไบต์