**Homework #1** Due: Oct. 14

1. **What is the function of the CPU subsystem in a computer?**

The CPU consists of three parts:

1. ALU (Arithmetic Logic Unit) – Carries out Arithmetical and Logical Operations

2. Control Unit – Controls the activities of the CPU

3. Registers – Memory within the CPU

It is the electronic circuitry that carries out the instructions of a computer program, by performing basic arithmetic, logic, controlling and I/O operations specified by the instructions. It fetches input from the RAM, processes it, sends back computed results.

1. **What is the difference between *von Neumann* and *Harvard* model?**

von Neumann is associated with using a single path to access a main memory that holds both instruction and data. Harvard is associated with having separate memories and access paths for instruction and data, allowing transfers to be performed at the same time.

1. **Convert the hexadecimal number A9 to:**
2. **octal**

A916  = A 9 base 16

= 10 9 base 16

= 1010 1001 base 2

= 10 101 001

= 2 5 1` base 8

= 2518

1. **decimal**

A916  = A \* 16^1 + 9 \* 16^0

= 10 \* 16^1 + 9 \* 16^0

= 160 + 9

= 16910

1. **binary**

A916  = A 9

= 10 9

**=** 101010012

1. **Convert the decimal number 169 to:**
2. **binary**

16910  = 16910

* 84 1
* 42 0
* 21 0
* 10 1
* 5 0
* 2 1
* 1 0

= 101010012

1. **octal**

16910  = 169­10

* 21 1
* 2 5
* 2 2

= 2518

1. **hexadecimal**

16910  = 16910

* 10 9
* 10 A

= A916

1. **Convert the binary number 10101001 to:**
2. **octal**

101010012  = 101010012

= 10 101 001

= 2518

1. **decimal**

101010012  = 101010012

= 1 \* 2^7 + 0 + 1\* 2^5 + 0 + 1\* 2^3 + 0 + 0 + 1 \* 2^0

= 1 + 8 + 32 + 128

= 16910

1. **hexadecimal**

101010012  = 101010012

= 1010 1001

= A 9

= A916

1. **What is ASCII code? What is the ASCII code for the character 5?**

ASCII stands for American Standard Code Information Interchange. It’s a 7-bit (128) character code where every single bit represents a unique character. There is also an extended 8-bit (256) version, with more symbols.

ASCII code for the character 5 = 01101012

= 0658

= 5310

= 3516

1. **Use 8 bits memory to represent integer number 5.**

510  = 510

=> 2 1

=> 1 0

= 1012

= 000001012 8-bit representation

1. **Use two’s complement method to store the following decimal integers to binary numbers with 8-bit allocation.**
2. **53**

5310  = 53 base10

=> 26 1

=> 13 0

=> 6 1

=> 3 0

=> 1 1

= 1101012

= 001101012 8-bit representation

Two’s complement = One’s complement + 1

=> One’s complement of 001101012

=> 110010102

Add 1

=> 110010112

Two’s complement of 001101012 = 110010112

1. **−107**

-107 base10 = - (+107) base10

=> - (53) 1

=> - (26) 1

=> - (13) 0

=> - (6) 1

=> - (3) 0

=> - (1) 1

= 111010112 Added because of the negative sign

Two’s complement = One’s complement +1

=> One’s complement of 111010112

=> 000101002

Add 1

=> 000101012

Two’s complement of 111010112 = 000101012

1. **Normalize the following binary floating-point numbers. Explicitly show the value of the exponent after normalization.**
2. **(101.11)2**

101.112  = 1 \* 2^2 + 0 + 1 \* 2^0 + 1 \* 2^-1 + 1 \* 2^-2

= 4 + 0 + 1 + 0.5 + 0.25

= 5.7510

1. **(10.111)2**

10.1112 = 1\* 2^1 + 0 + 1 \* 2^-1 + 1 \* 2^-2 + 1 \* 2^-3

= 2 + 0.5 + 0.25 + 0.125

= 2.87510

1. **Convert the decimal number to binary number:**
2. **161.875**

161.87510 = 161.87510

=> 161

=> 80 1

=> 40 0

=> 20 0

=> 10 0

=> 5 0

=> 2 1

=> 1 0

=> .875

=> 1.75 1

=> .75

=> 1.5 1

=> .5

=> 1

161.87510 = 10100001.1112

1. **0.0234375**

0.023437510 = 0.023437510

=> 0.046875 0

=> 0.09375 0

=> 0.1875 0

=> 0.375 0

=> 0.75 0

=> 1.5 1

=> .5

=> 1

0.023437510 = .00000112