Information Technology Workshop-1 (ITWS1)

Instructor - Shiv Ram Dubey

Introduction

Overview of Discussion

What is computer?

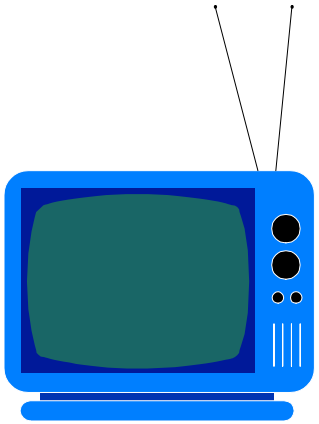
This Class -

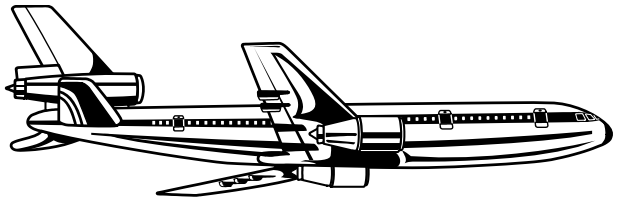
- What is a computer?
- What can computers do?
- What do computers understand?
- How to convert from Binary-Decimal-Hexa?

Next Class -

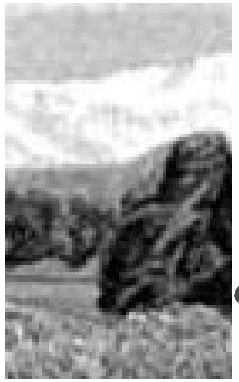
- What are the computer basics?
- How do computers solve problems?
- What is computer science?

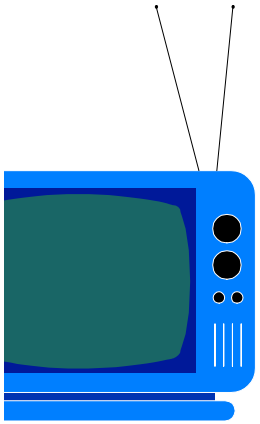
Which one is the computer?

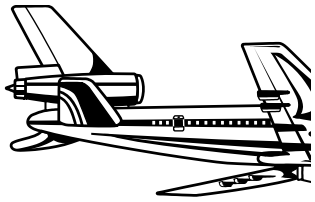




Which one is the computer?







Is a rock a computer?

Is a rock a computer?

- Does not act or process
- No input & no output

- Computers - handle *input* and *output*

Is washing machine a computer?

- Input: dirty clothes
- Output: clean clothes

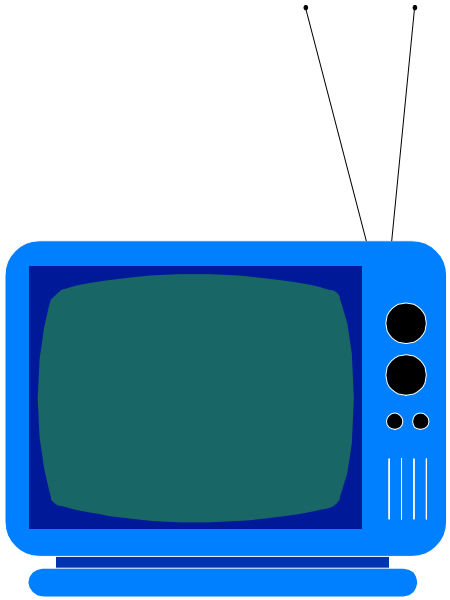
Is washing machine a computer?

- Input: dirty clothes
- Output: clean clothes

- Does not handle information

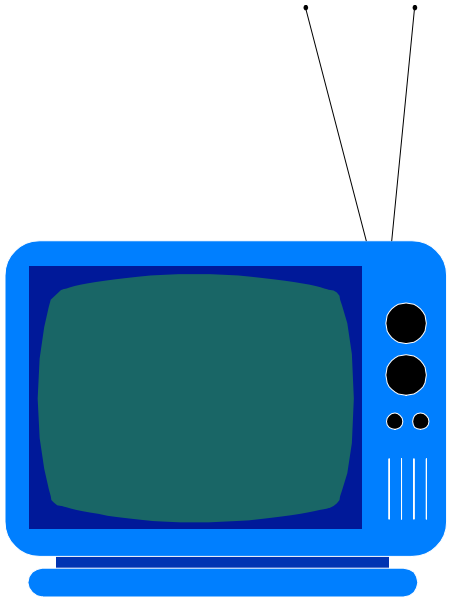
- Computers - input and output *information*

Is a television set a computer?



- Input: information from cables or radio waves
- Output: information as sound and picture

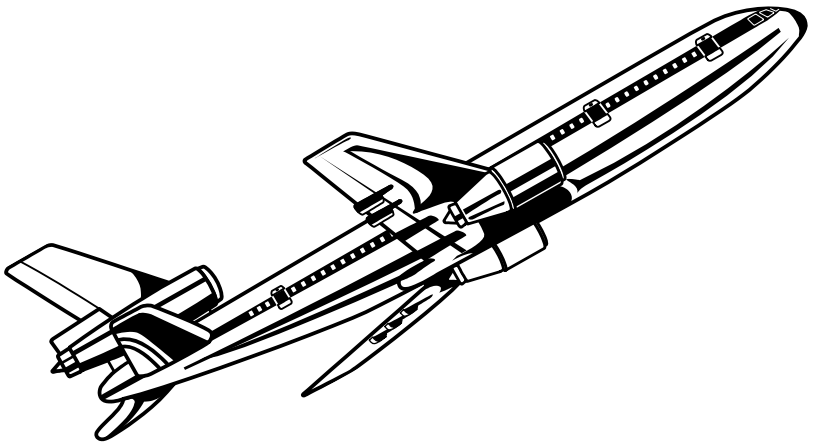
Is a television set a computer?



- Input: information from cables or radio waves
- Output: information as sound and picture
- Does not process information by computing it

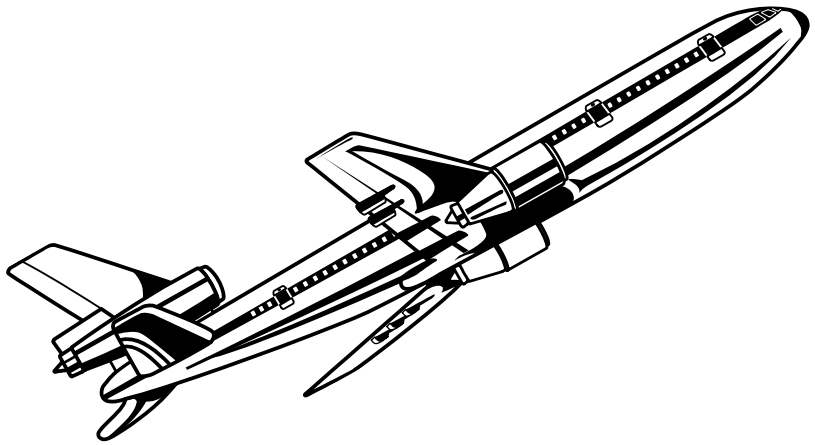
- Computers *process* information by computing new results

Is a modern airplane a computer?



- Input: information from radio waves
- Output: manipulations to the airplane

Is a modern airplane a computer?



- Input: information from radio waves
- Output: manipulations to the airplane
- Handles specific information

- Computers are *general purpose*

Is ordinary calculator a computer?

- Input: numbers and mathematical operations
- Output: answer
- Handles any numeric task

Is ordinary calculator a computer?

- Input: numbers and mathematical operations
- Output: answer
- Handles any numeric task
- Cannot be programmed

- Computers are *programmable*

Definition of a Computer

Definition of a Computer

General purpose,

Programmable,

Computing the information

With input and output

What can computers do – today?

What can computers do – today?

Business productivity managers

Personal information managers

Spreadsheets

Database software

Desktop publishing

Multimedia encyclopedias

Simulate the physical world

Produce a music video

And many more ……

What might computers do–tomorrow?

What might computers do–tomorrow?

Diagnose diseases

Control robots that walk, talk, and learn

Compose music and create art

Information forensics

Artificial intelligence

And many more .....

What do computers understand?

Binary

0/1

True/False

On/Off

Decimal Notation

Decimal Notation

123

100	10	1	- Places
1	2	3	

$123 = 1 \times 100 + 2 \times 10 + 3 \times 1$

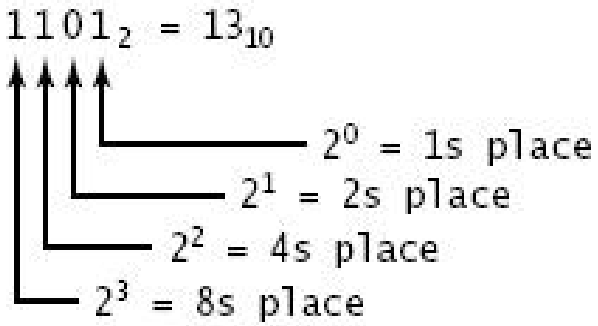
Binary Notation

Binary Notation

123 = 01111011

Binary to Decimal

Binary to Decimal



Binary to Decimal

Binary Number	Decimal Equivalent
11_2 →	1*2 + 1*1 = 3
1101_2 →	1*8 + 1*4 + 1*2 + 1*1 = 13
10011_2 →	1*16 + 0*8 + 0*4 + 1*2 + 1*1 = 19
100110_2 →	1*32 + 0*16 + 0*8 + 1*4 + 1*2 + 0*1 = 38

Decimal to Binary

Decimal to Binary

Converting 19 to binary:

```
 19 is odd   → B =     1,   D = Math.floor(19/2) = 9
  9 is odd   → B =    11,   D = Math.floor(9/2) =  4
  4 is even  → B =   011,   D = Math.floor(4/2) =  2
  2 is even  → B =  0011,   D = Math.floor(2/2) =  1
  1 is odd   → B = 10011,   D = Math.floor(1/2) =  0
```

Converting 116 to binary:

```
116 is even  → B =     0,   D = Math.floor(116/2) = 58
 58 is even  → B =    00,   D = Math.floor(58/2) = 29
 29 is odd   → B =   100,   D = Math.floor(29/2) = 14
 14 is even  → B =  0100,   D = Math.floor(14/2) = 7
  7 is odd   → B = 10100,   D = Math.floor(7/2) = 3
  3 is odd   → B =110100,   D = Math.floor(3/2) = 1
  1 is odd   → B =1110100,  D = Math.floor(1/2) = 0
```

Hexadecimal: The 16 Hex Digits

0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

A = 10, B = 11, ... , F = 15

0101 1100

5 C

Binary-HEX

- Example:
- $(01000010101001010110111110001)_2 = (?)_{16}$

Binary-HEX

- Example:
- $(01000010101001010110111110001)_2 = (?)_{16}$
- 0100 0010 1010 0101 0110 1111 0001
- $= (4\ 2\ A\ 5\ 6\ F\ 1)_{16}$

HEX-Binary

- Example:
- $(AF52C)_{16} = (?)_2$

HEX-Binary

- Example:
- $(AF52C)_{16} = (?)_2$
- A F 5 2 C
- $= (1010\ 1111\ 0101\ 0010\ 1100)_2$

Representing Integers

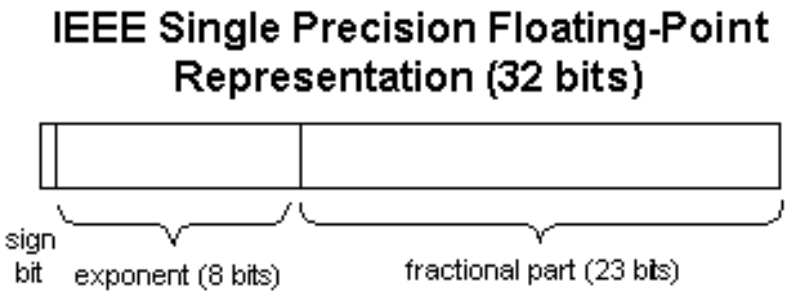
Bit Pattern	*Decimal Value*
10000000000000000000000000000000	(-2^{31} = -2,147,483,648)
10000000000000000000000000000001	($-2^{31}-1$ = -2,147,483,647)
10000000000000000000000000000010	($-2^{31}-2$ = -2,147,483,646)
.	
.	
.	
11111111111111111111111111111101	(-3)
11111111111111111111111111111110	(-2)
11111111111111111111111111111111	(-1)
00000000000000000000000000000000	(0)
00000000000000000000000000000001	(1)
00000000000000000000000000000010	(2)
00000000000000000000000000000011	(3)
.	
.	
.	
01111111111111111111111111111101	($2^{31}-3$ = 2,147,483,645)
01111111111111111111111111111110	($2^{31}-2$ = 2,147,483,646)
01111111111111111111111111111111	($2^{31}-1$ = 2,147,483,647)

Representing Real Numbers

$123.45 = 12345 \times 10^{-2}$ $.000042 = 42 \times 10^{6}$

Representing Real Numbers

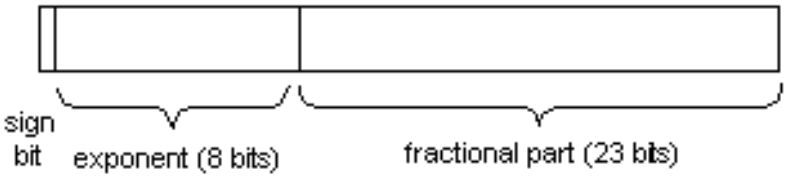
123.45 = 12345 x 10^{-2} .000042 = 42 x 10^{6}



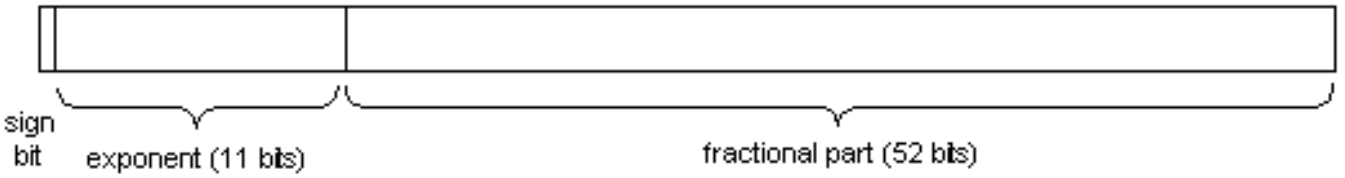
Representing Real Numbers

123.45 = 12345 x 10^{-2} .000042 = 42 x 10^{6}

IEEE Single Precision Floating-Point Representation (32 bits)



IEEE Double Precision Floating-Point Representation (64 bits)



ASCII

(American Standard Code for Information Interchange)

A – 65	a – 97
B – 66	b – 98
C – 67	c – 99
.	.
.	.
.	.

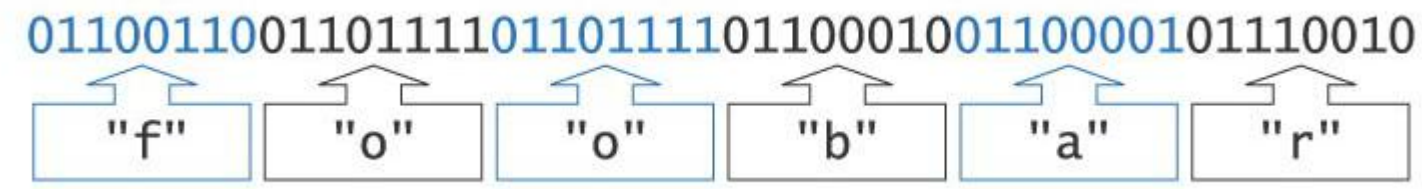
http://www.asciitable.com/

Representing Characters

ASCII Character Codes

code	char	code	char	code	char
00100000	space	01000000	@	01100000	'
00100001	!	01000001	A	01100001	a
00100010	"	01000010	B	01100010	b
00100011	#	01000011	C	01100011	c
00100100	$	01000100	D	01100100	d
00100101	%	01000101	E	01100101	e
00100110	&	01000110	F	01100110	f
00100111	'	01000111	G	01100111	g
00101000	(	01001000	H	01101000	h
00101001	)	01001001	I	01101001	i
00101010	*	01001010	J	01101010	j
00101011	+	01001011	K	01101011	k
00101100	,	01001100	L	01101100	l
00101101	-	01001101	M	01101101	m
00101110	.	01001110	N	01101110	n
00101111	/	01001111	O	01101111	o
00110000	0	01010000	P	01110000	p
00110001	1	01010001	Q	01110001	q
00110010	2	01010010	R	01110010	r
00110011	3	01010011	S	01110011	s
00110100	4	01010100	T	01110100	t
00110101	5	01010101	U	01110101	u
00110110	6	01010110	V	01110110	v
00110111	7	01010111	W	01110111	w
00111000	8	01011000	X	01111000	x
00111001	9	01011001	Y	01111001	y
00111010	:	01011010	Z	01111010	z
00111011	;	01011011	[	01111011	{
00111100	<	01011100	\	01111100	|
00111101	=	01011101	]	01111101	}
00111110	>	01011110	^	01111110	~
00111111	?	01011111	_	01111111	delete

Representing Text



Representing Text

- The size of a file = number of bytes stored in the file
- 1 KB = ? bytes

Representing Text

- The size of a file = number of bytes stored in the file
- 1 KB = 1024 bytes = 2^{10} bytes

Representing Text

- The size of a file = number of bytes stored in the file

- 1 KB = 1024 bytes = 2^{10} bytes
- 1 MB = 1024 KB = 2^{20} bytes
- 1 GB = 1024 MB = 2^{30} bytes
- 1 TB = 1024 GB = 2^{40} bytes

Syllabus

Introduction to Computers

HTML & CSS

Unix Commands

Latex, Beamer

JavaScript and Game Development

PHP

Bash Scripting

Grading

10% Exam 1

10% Exam 2

25% Exam 3

15% Quiz

40% Lab Assignments + Projects

Course Website

https://sites.google.com/site/shivram1987/teaching/f17_itws1