EE 101 -Introduction to Programming

Mehmet Bozdal

ABDULLAH GÜL ÜNİVERSİTESİ

About me

Email: mehmet.bozdal@agu.edu.tr

Office: F0A11

Office Hours: to be announced

Organization of the Course

- The course has following components:
- Lecture (https://bit.ly/aguee101)
- Laboratory
- Integrated Homework
- Integrated Quiz
- Subcomponent Quiz
- Project

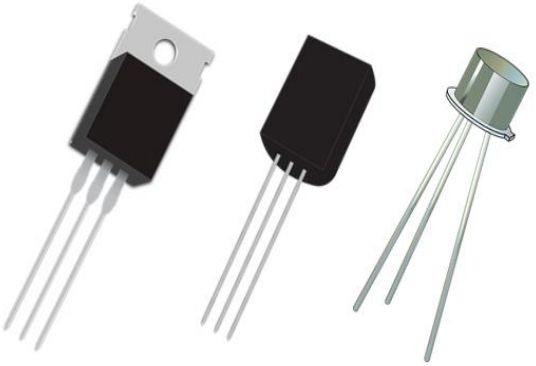
What will you learn?

- Basics of algorithm
- Programming C++
- Basics of object-oriented programming

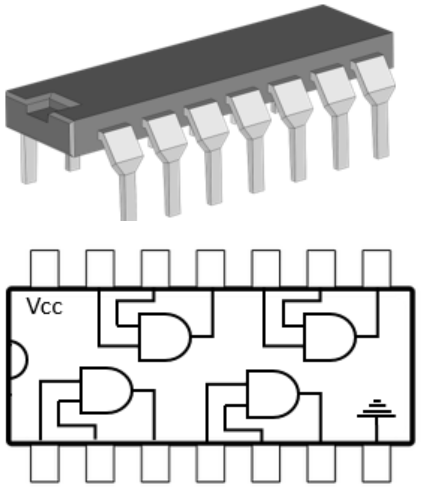
- Course Book: **C++ How to Program**, Harvey Deitel and Paul Deitel

It is all about 1’s and 0’s

Vacuum Tube (1904)



Transistors (1947)



7408: Quad 2-input AND gate

Processors

American Standard Code for Information Interchange(ASCII)

Dec	Hex	Oct	Binary	Char	Dec	Hex	Oct	Binary	Char	Dec	Hex	Oct	Binary	Char	Dec	Hex	Oct	Binary	Char
0	00	000	0000000	NUL (null character)	32	20	040	0100000	space	64	40	100	1000000	@	96	60	140	1100000	`
1	01	001	0000001	SOH (start of header)	33	21	041	0100001	!	65	41	101	1000001	A	97	61	141	1100001	a
2	02	002	0000010	STX (start of text)	34	22	042	0100010	"	66	42	102	1000010	B	98	62	142	1100010	b
3	03	003	0000011	ETX (end of text)	35	23	043	0100011	#	67	43	103	1000011	C	99	63	143	1100011	c
4	04	004	0000100	EOT (end of transmission)	36	24	044	0100100	$	68	44	104	1000100	D	100	64	144	1100100	d
5	05	005	0000101	ENQ (enquiry)	37	25	045	0100101	%	69	45	105	1000101	E	101	65	145	1100101	e
6	06	006	0000110	ACK (acknowledge)	38	26	046	0100110	&	70	46	106	1000110	F	102	66	146	1100110	f
7	07	007	0000111	BEL (bell (ring))	39	27	047	0100111	'	71	47	107	1000111	G	103	67	147	1100111	g
8	08	010	0001000	BS (backspace)	40	28	050	0101000	(	72	48	110	1001000	H	104	68	150	1101000	h
9	09	011	0001001	HT (horizontal tab)	41	29	051	0101001	)	73	49	111	1001001	I	105	69	151	1101001	i
10	0A	012	0001010	LF (line feed)	42	2A	052	0101010	*	74	4A	112	1001010	J	106	6A	152	1101010	j
11	0B	013	0001011	VT (vertical tab)	43	2B	053	0101011	+	75	4B	113	1001011	K	107	6B	153	1101011	k
12	0C	014	0001100	FF (form feed)	44	2C	054	0101100	,	76	4C	114	1001100	L	108	6C	154	1101100	l
13	0D	015	0001101	CR (carriage return)	45	2D	055	0101101	-	77	4D	115	1001101	M	109	6D	155	1101101	m
14	0E	016	0001110	SO (shift out)	46	2E	056	0101110	.	78	4E	116	1001110	N	110	6E	156	1101110	n
15	0F	017	0001111	SI (shift in)	47	2F	057	0101111	/	79	4F	117	1001111	O	111	6F	157	1101111	o
16	10	020	0010000	DLE (data link escape)	48	30	060	0110000	0	80	50	120	1010000	P	112	70	160	1110000	p
17	11	021	0010001	DC1 (device control 1)	49	31	061	0110001	1	81	51	121	1010001	Q	113	71	161	1110001	q
18	12	022	0010010	DC2 (device control 2)	50	32	062	0110010	2	82	52	122	1010010	R	114	72	162	1110010	r
19	13	023	0010011	DC3 (device control 3)	51	33	063	0110011	3	83	53	123	1010011	S	115	73	163	1110011	s
20	14	024	0010100	DC4 (device control 4)	52	34	064	0110100	4	84	54	124	1010100	T	116	74	164	1110100	t
21	15	025	0010101	NAK (negative acknowledge)	53	35	065	0110101	5	85	55	125	1010101	U	117	75	165	1110101	u
22	16	026	0010110	SYN (synchronize)	54	36	066	0110110	6	86	56	126	1010110	V	118	76	166	1110110	v
23	17	027	0010111	ETB (end transmission block)	55	37	067	0110111	7	87	57	127	1010111	W	119	77	167	1110111	w
24	18	030	0011000	CAN (cancel)	56	38	070	0111000	8	88	58	130	1011000	X	120	78	170	1111000	x
25	19	031	0011001	EM (end of medium)	57	39	071	0111001	9	89	59	131	1011001	Y	121	79	171	1111001	y
26	1A	032	0011010	SUB (substitute)	58	3A	072	0111010	:	90	5A	132	1011010	Z	122	7A	172	1111010	z
27	1B	033	0011011	ESC (escape)	59	3B	073	0111011	;	91	5B	133	1011011	[	123	7B	173	1111011	{
28	1C	034	0011100	FS (file separator)	60	3C	074	0111100	<	92	5C	134	1011100	\	124	7C	174	1111100	|
29	1D	035	0011101	GS (group separator)	61	3D	075	0111101	=	93	5D	135	1011101	]	125	7D	175	1111101	}
30	1E	036	0011110	RS (record separator)	62	3E	076	0111110	>	94	5E	136	1011110	^	126	7E	176	1111110	~
31	1F	037	0011111	US (unit separator)	63	3F	077	0111111	?	95	5F	137	1011111	_	127	7F	177	1111111	DEL

Dec	Hex	Oct	Binary	Char	Dec	Hex	Oct	Binary	Char
64	40	100	1000000	@	96	60	140	1100000	`
65	41	101	1000001	A	97	61	141	1100001	a
66	42	102	1000010	B	98	62	142	1100010	b
67	43	103	1000011	C	99	63	143	1100011	c
68	44	104	1000100	D	100	64	144	1100100	d
69	45	105	1000101	E	101	65	145	1100101	e

ASCII Code

- Hello AGU!
- 072 101 108 108 111 032 065 071 085 033

- EE 101
- 069 069 032 049 048 049

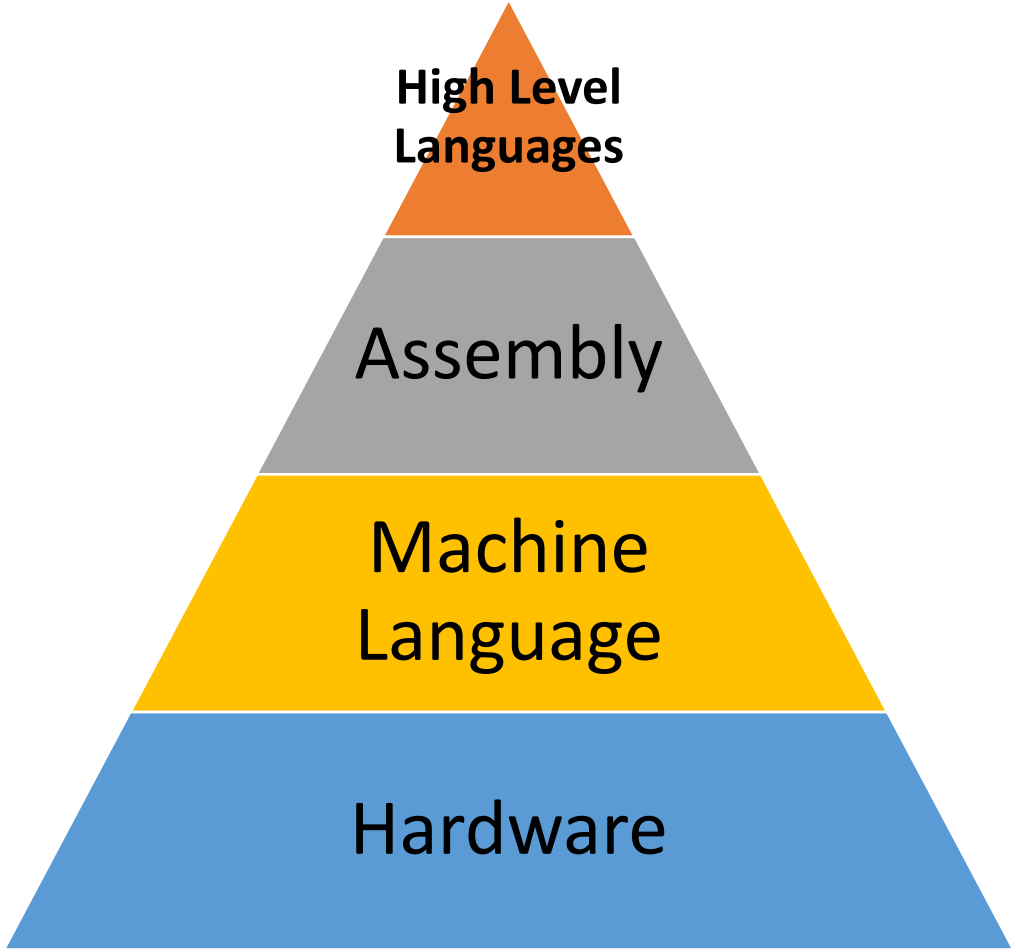
How to represent colour or images?

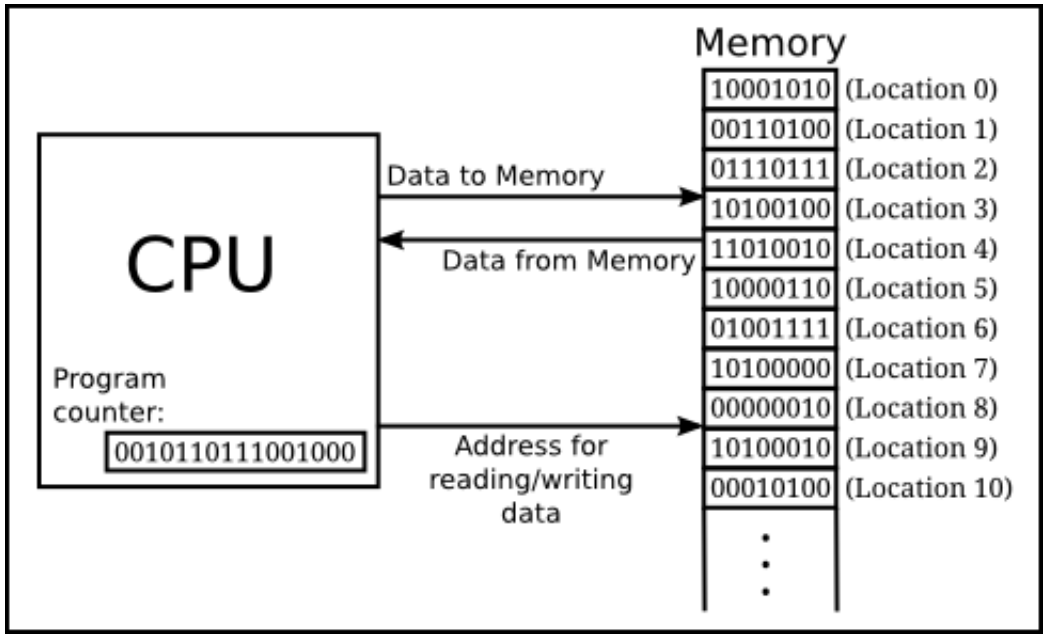
- rgb(RED , GREEN , BLUE)
  - Rgb(255,0,0)
  - Rgb(0,0,255)
  - Rgb(255,100,255)
- Portable Pix (or pixel) Map

cake.ppm

```
P3
720 540
255
123 125 124
124 126 125 127 129 128 129 131 130 128 130 129 126 128 127
132 134 133 132 135 134 135 140 139 137 143 141 136 142 140 135 143 140
134 144 140 135 147 142 138 145 143 139 143 142 138 142 141 137 141 140
139 144 143 139 143 142 139 143 142 140 143 141 143 142 140 140 140 137
136 137 135 136 137 134 137 139 136 137 139 136 138 140 137 140 142 137
144 146 141 143 145 140 139 141 135 137 139 134 134 136 131 137 139 136
139 143 142 140 144 143 141 145 144 141 146 145 144 148 147
146 149 148 145 150 149 146 151 147 148 153 149 149 155 151 148 155 151
146 155 150 147 157 152 146 157 152 148 155 152 148 154 152 146 155 152
146 155 152 148 159 155 148 159 155 148 160 155 150 160 157 153 159 157
150 155 154 149 155 153 152 158 156 154 160 158 154 160 158 152 158 156
152 158 158 149 159 157 147 159 157 147 161 159 149 161 160 152 160 160
154 159 160 156 162 158 157 163 159 155 161 157 157 163 159
158 164 160 159 164 160 159 165 161 156 165 162 159 169 166 158 167 164
158 167 164 157 166 163 156 165 162 157 167 164 160 167 165 161 167 165
158 166 163 157 166 163 158 169 165 161 172 168 159 171 166 157 169 165
159 170 166 160 169 166 163 170 168 163 169 167 163 168 166 164 167 166
162 167 166 160 169 166 161 170 167 162 171 168 163 172 169 162 171 168
162 171 168 163 172 169 168 171 170 170 172 172 170 173 172
167 172 171 165 171 169 166 172 170 164 171 169 161 172 168 161 172 168
161 170 167 163 169 167 166 168 167 168 169 168 168 166 167 165 169 168
164 172 170 167 174 171 168 174 172 169 173 172 171 173 173 173 175 174
```

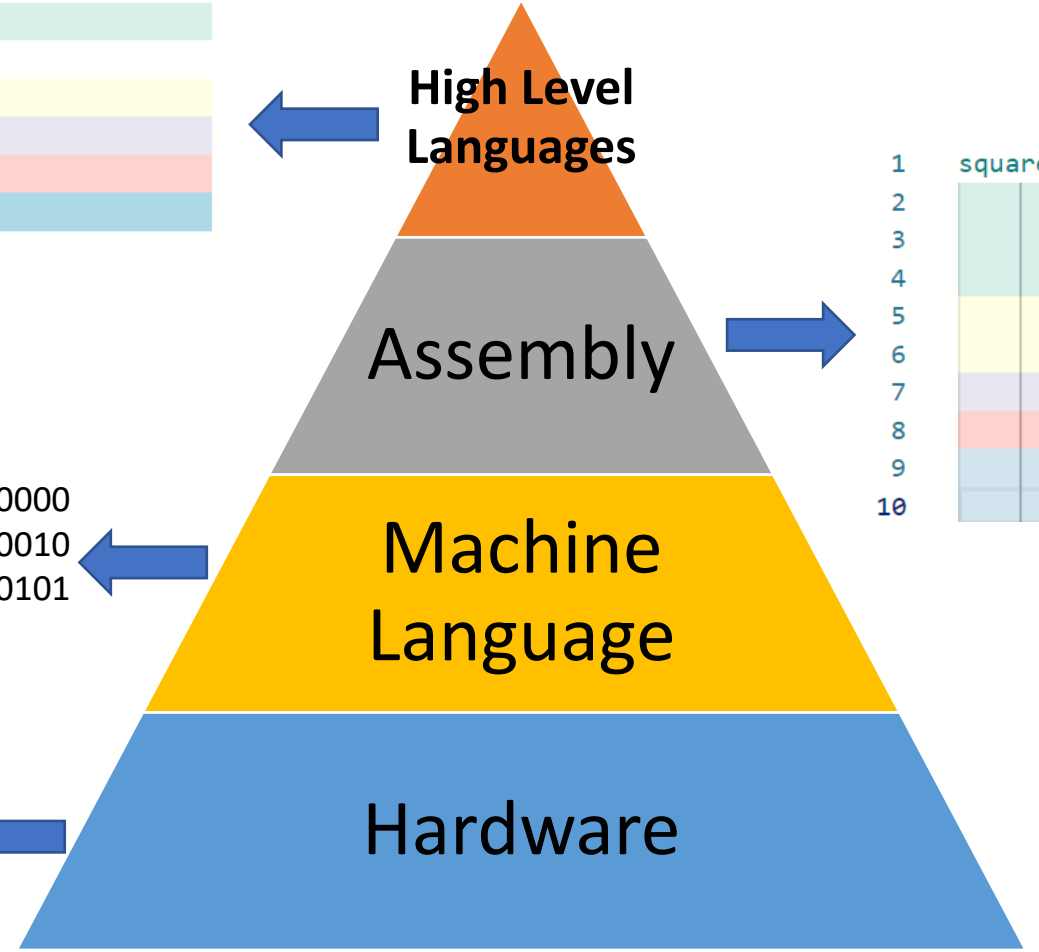
How does computer work?





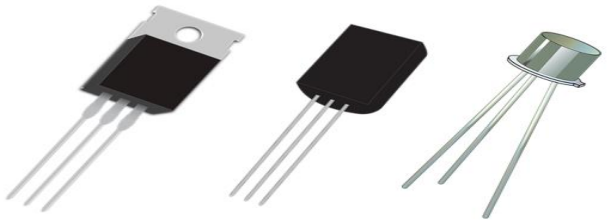
Programming Languages

```
1   // Type your code here, or load an example.
2   int square(int num) {
3
4       int x=0,y=0;
5       x=100;
6       return x;
7   }
```




```
1   square(int):
2           push    rbp
3           mov     rbp, rsp
4           mov     DWORD PTR [rbp-20], edi
5           mov     DWORD PTR [rbp-4], 0
6           mov     DWORD PTR [rbp-8], 0
7           mov     DWORD PTR [rbp-4], 100
8           mov     eax, DWORD PTR [rbp-4]
9           pop     rbp
10          ret
```

01001000 01100101 01101100 01101100 01101111 00100000
01100011 01101100 01100101 01110110 01100101 01110010
00100000 01110011 01110100 01110101 01100100 01100101
01101110 01110100 00100000 00111010 00101001



Compiler

Source Code

```cpp
#include <iostream>

int main() {
    std::cout << "Hello, World!" << std::endl;
    return 0;
}
```

Machine Code

01001000 01100101 01101100 01101100 01101111 00100000
01100011 01101100 01100101 01110110 01100101 01110010
00100000 01110011 01110100 01110101 01100100 01100101
01101110 01110100 00100000 00111010 00101001

Q&A

Thanks…