

CS01

2020-2021

Tanışalım

Haftalık Plan

Hafta 1	Tanışma & Bilgisayar Tarihçesi
Hafta 2	Bilgisayar Bilimi & Soyutlama & İşletim Sistemleri
Hafta 3	Bellek Nedir & C'de ilk programlar
Hafta 4	Python: Merhaba dünya, koşullar, döngüler ..
Hafta 5	Python: Arrayler, dictionaryler, recursion
Hafta 6	Python: Nesneye yönelik programlama
Hafta 7	Python: Veri Yapıları; Linked list, tree
Hafta 8	Web sistemleri ve Veri Tabanları
Hafta 9	IOT, Cloud ve Sanal Makineler
Hafta 10	Günümüz, Açık kaynak ve özgür yazılım

Sizden Beklenenler

Ödevlerin eksiksiz yapılması ve tüm derslere katılım.



Ders Formatı

- Derslerde öğrencilerin programlama yapmasına gerek olmayacaktır
- Ders sonrasında öğrenciler ödevleri eksiksiz bir şekilde yapacaklardır



Ödevler

- Ödevlerin yapılması zorunludur. Dersin tam olarak idrak edilebilmesi için verilen okumaların videoların izlenmesi hatta gerekirse birden çok tekrarlanması güzel olacaktır.
- Ödevler
 - Alıştırmalar
 - Okumalar
 - Videolar

Ödevler

musaunal.com/cs01/



CS01- Programlamaya Giriş



Programlamaya giriş dersine hoşgeldiniz. Bu ders kapsamında, bilgisayar mühendisliği ve bilimi için güzel bir başlangıç yapacak ve ileride öğreneceğiniz konular için önemli bir temel atmuş olacaksınız.

- Ders sonucu olarak bir sınav **yapılmayacaktır**.
- Ödevlerin yapılması **zorunludur**.
- Ders puan notu katılım ve ödevlere göre değerlendirilecektir.
- Ders hakkında sorularınızı umusasadi@ gmail . com ' a yazabilirsiniz

[Ders drive dosyası](#)

~ Haftalık Plan

Hafta	Ders	Ders Saati
1	Tanışma & Bilgisayar Tarihi	3
2	Bilgisayar Bilimi & Soyutlama & İşletim Sistemleri	3
3	Bellek Nedir & C'de ilk programlar	3
4	Python: Merhaba dünya, koşullar, döngüler ..	3
5	Python: Arrayler, dictionaryler, recursion	3
6	Python: Nesneye yönelik programlama	3
7	Python: Veri Yapıları; Linked list, tree	4
8	Web sistemleri ve Veri Tabanları	4
9	IOT, Cloud ve Sanal Makineler	5

8	Web sistemleri ve Veri Tabanları	4
9	IOT, Cloud ve Sanal Makineler	5
10	Günümüz, Açık kaynak ve özgür yazılım	4

~ Ödevler

- 📁 Ödev 1

~ Okumalar

- 📖 Hafta 1 - A Brief History of Computers
- 📖 Hafta 1 - History of Operating Systems

~ Videolar

- 🎥 Hafta 1 - The mind behind Linux
- 🎥 Hafta 1 - 8 bitlik breadboard bilgisayarı programlama
- 🎥 Hafta 2 - Abstraction Layers from the Atom to the Data Center
- 🎥 Hafta 2 - Operating System Basics



Hesaplamalı Düşünme - Algoritmik Düşünme Nedir

Computational Thinking

**Problemi Küçük
Parçalara Ayır**

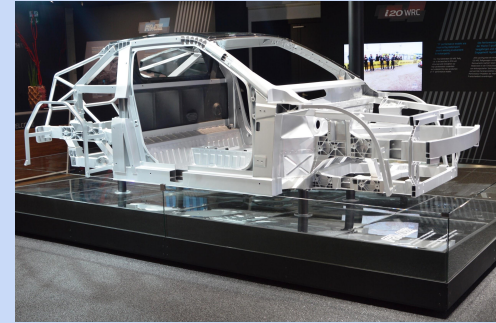
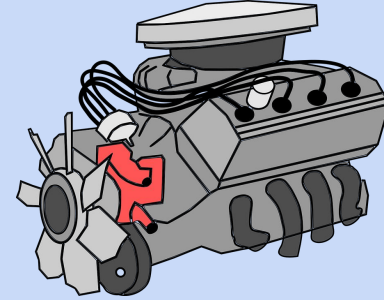
Soyutlama

Hesaplamalı Düşünme

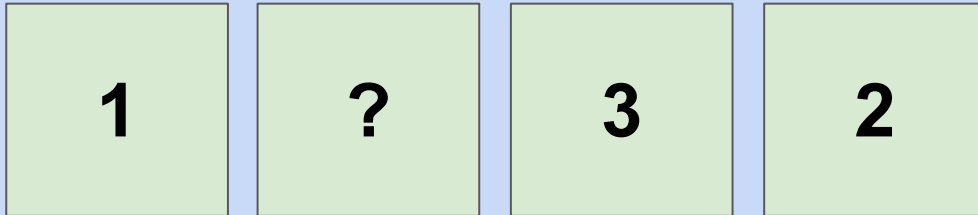
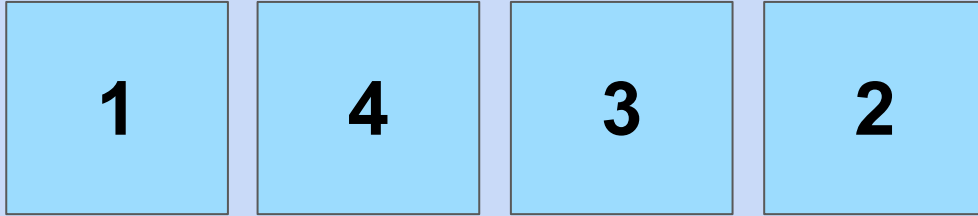
**BENZER PROBLEMLERİ
TANI**

**ÇÖZÜM YOLU -
ALGORİTMA**

Problemi Küçük Parçalara Ayır

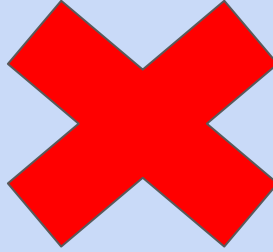


Benzer Problemleri TANI

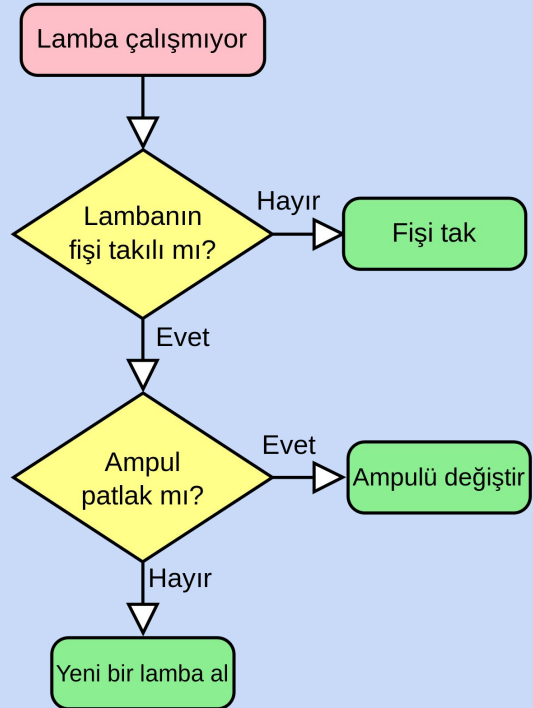


Problemi Kolaylařtır

- Kafa
- Ayaklar
- Gvde
- Kuyruk
- Sesi
- Sevdđđi Yiyecekler



Çözüm Yolu - Algoritma



Pseudocode

Eğer Kapı Açıksa:

İçeri Gir

Değilse:

Markete Git

Eğer İndirim Varsa:

Ürünleri Al

Yoksa:

İndirim Olan Vakti Bekle

Pseudocode

Bir Sayı Girin = sayi

Eğer sayi 25'den büyükse:

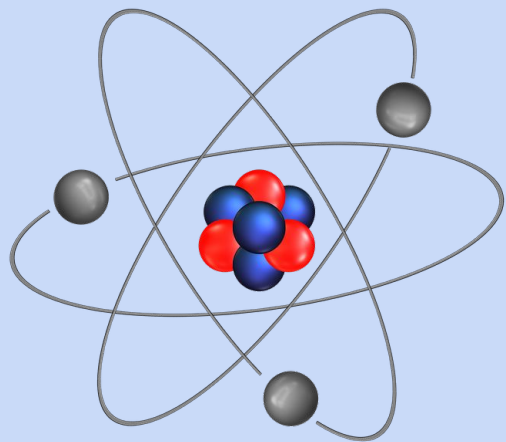
$\text{sayi} = \text{sayi} * 2$

Eğer sayi 25'den küçükse:

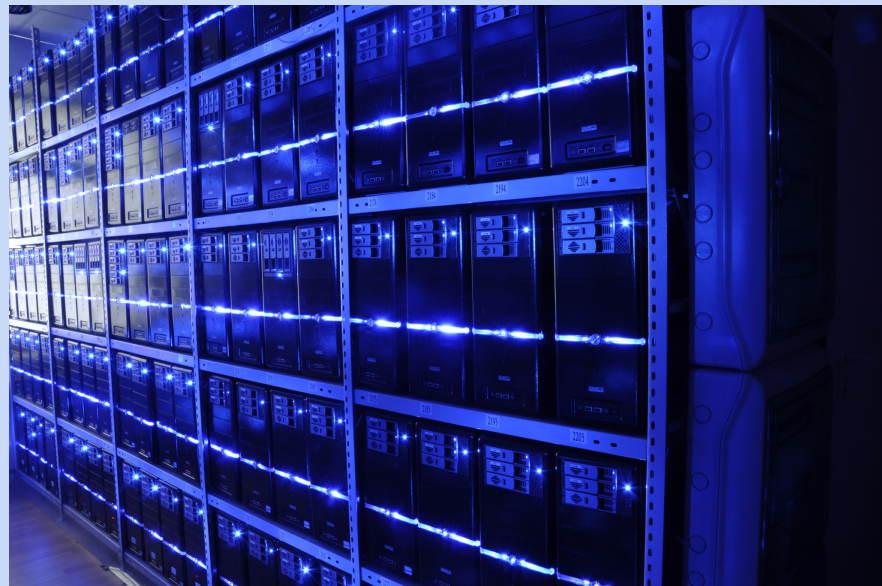
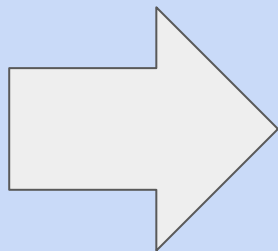
$\text{sayi} = \text{sayi} - 10$

Ekrana sayi'yi yazdır

SOYUTLAMA



?



Bilgisayar Dünyasının Kısa Tarihçesi - ?



- 10'luk sayı sistemi
- Çubuklar
- Ekinlerin sayılması
- Küçük ticaret anlaşmaları

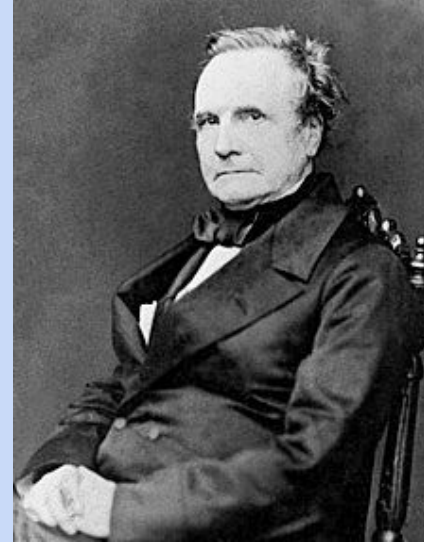
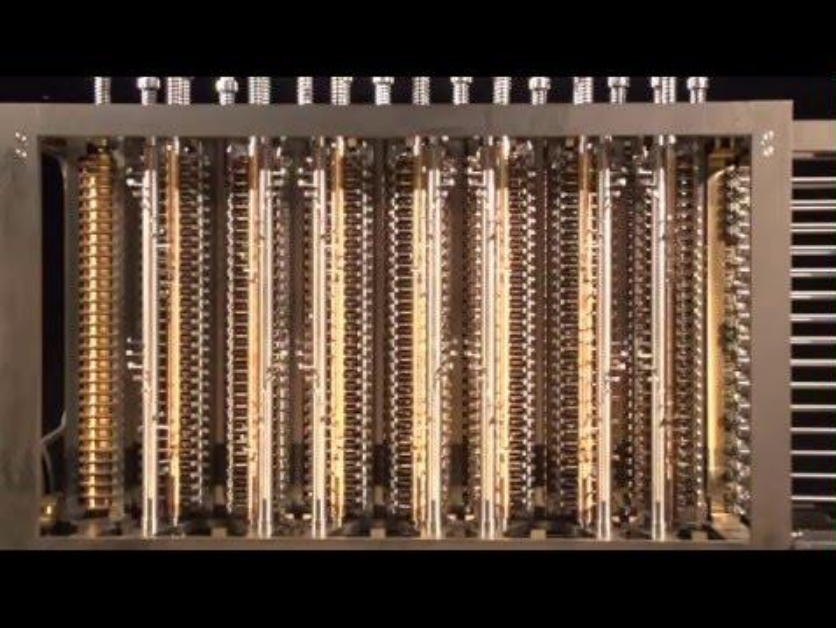
Bilgisayar Dünyasının Kısa Tarihçesi - Roma Dönemi

- Abaküs ile artık saymak daha kolay



Bilgisayar Dünyasının Kısa Tarihçesi - Fark makinesi

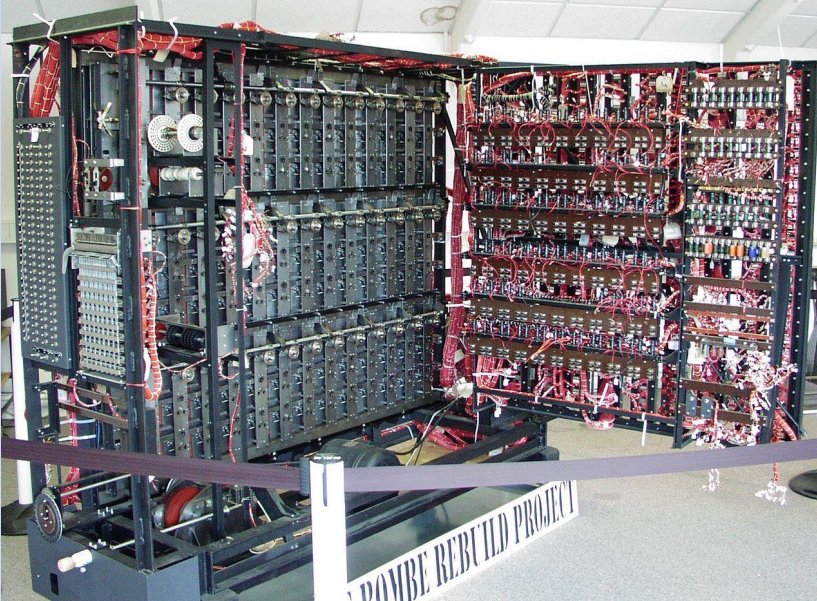
- Fark makinesi, polinom işlevlerin (fonksiyonların) hesaplanması için tasarlanmış bir mekanik hesap makinesidir. 30 haneye kadar doğru hesaplama yapabildiği görülmüştür.



Charles Babbage

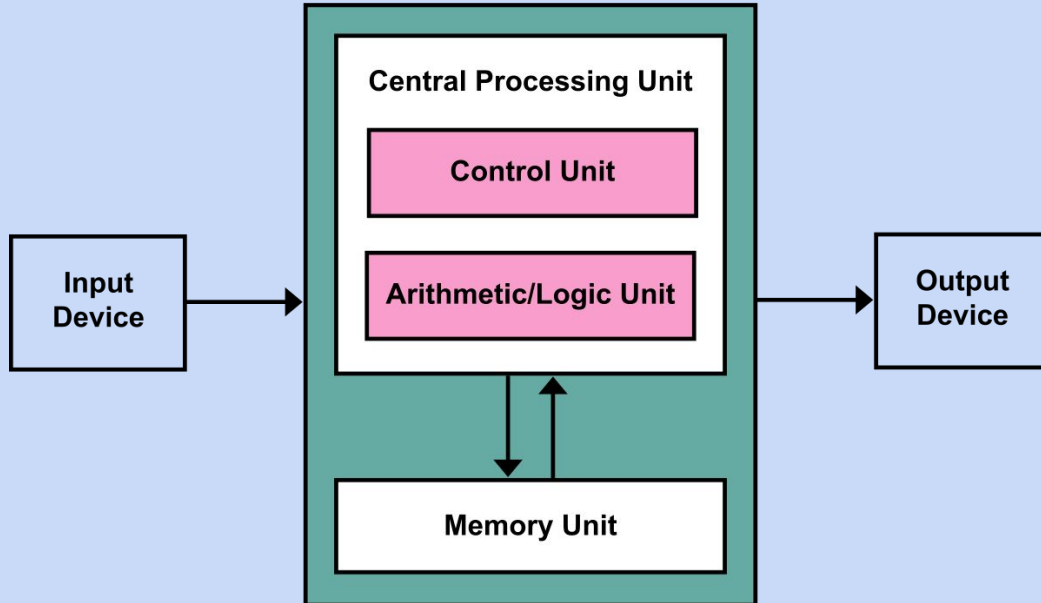
Bilgisayar Dünyasının Kısa Tarihçesi - İkinci Dünya Savaşı

- II. Dünya Savaşı sırasında Alman kriptu cihazlarının (Enigma) şifrelerinin kırılması için bir bilgisayar geliştirmiştir. Hesaplanabilir her şeyi hesaplayabilecek makinelerin doğuşu ..



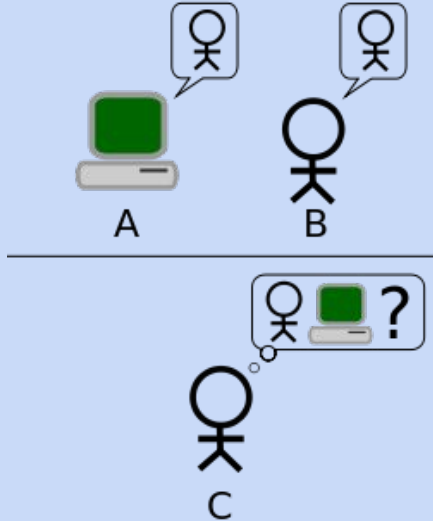
Bilgisayar Dünyasının Kısa Tarihçesi - Von Neumann

-

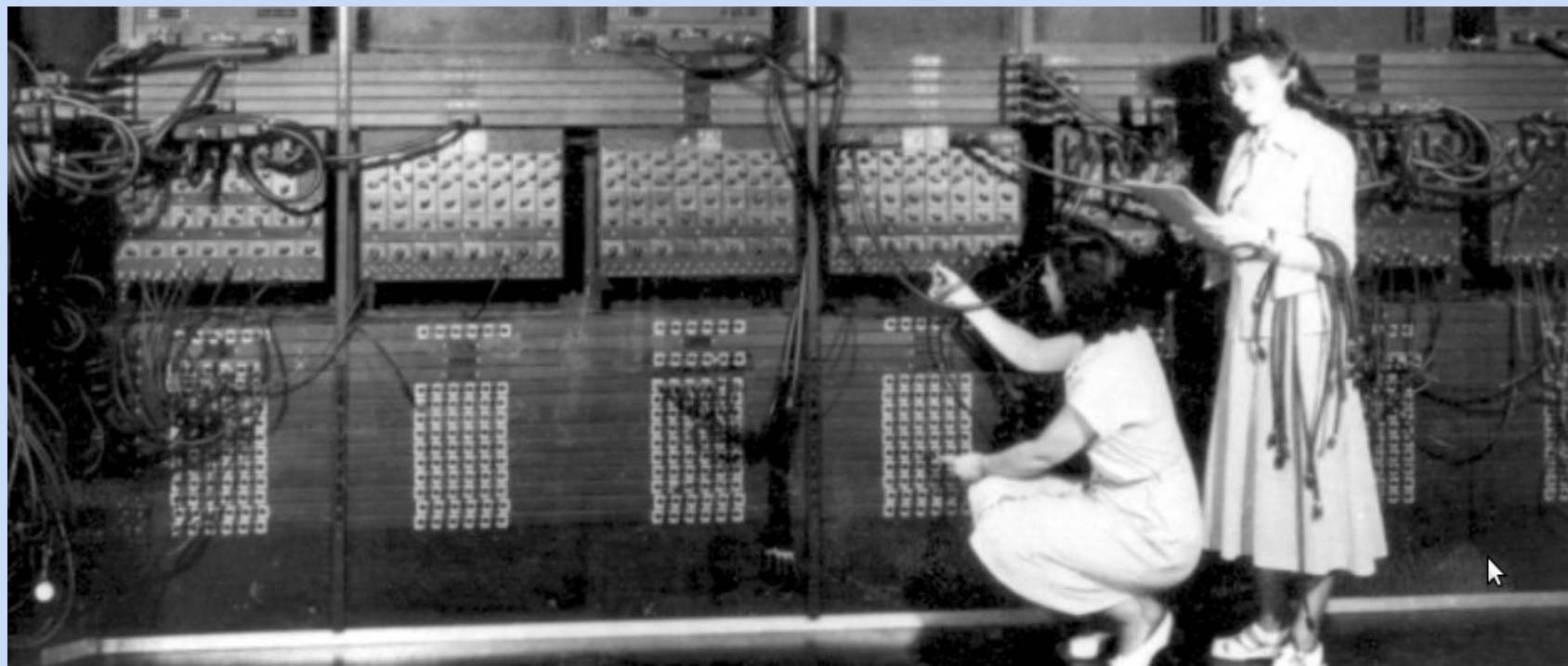


Bilgisayar Dünyasının Kısa Tarihçesi - Turing Testi

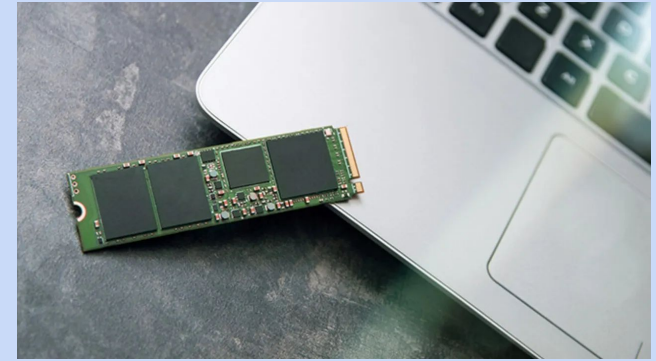
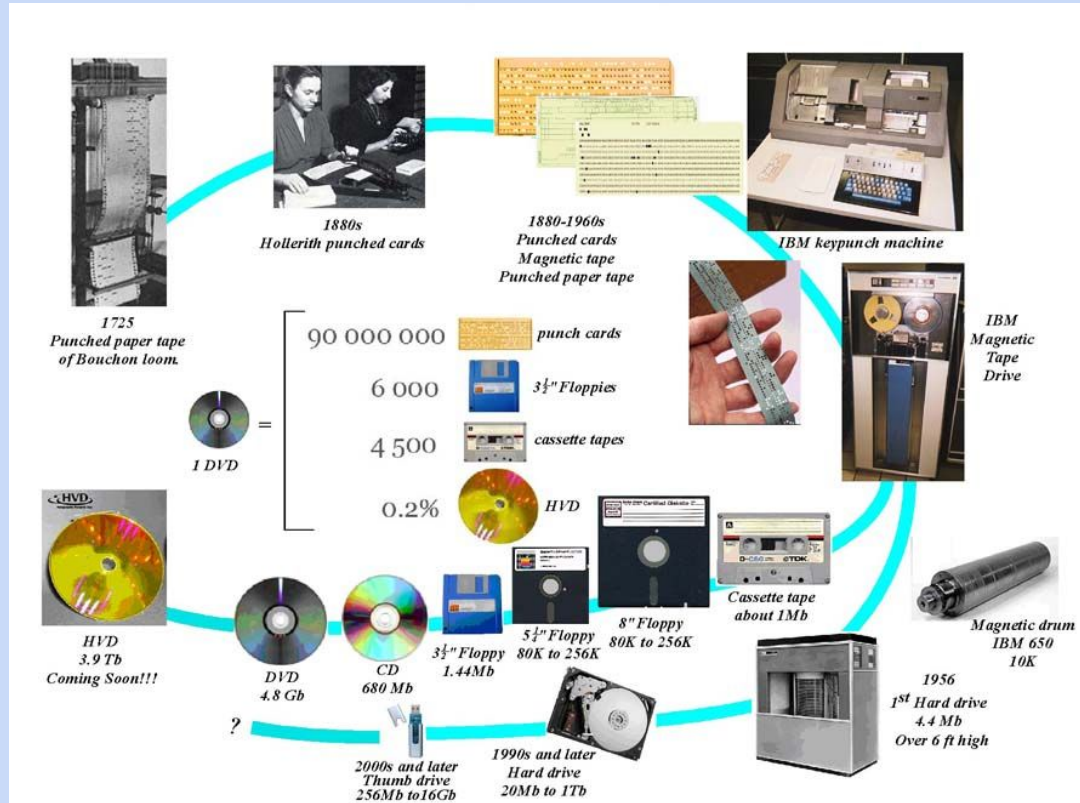
Bir makinenin düşünebildiğini söyleyebilmenin mantıksal olarak mümkün olup olmadığıdır. Bu test, yapısı itibarıyla bilgisayar bilimlerindeki pek çok yapay zeka probleminin çatısını teşkil etmektedir



Bilgisayar Dünyasının Kısa Tarihçesi - Yapay Zeka



Verilerimizi Nasıl Saklıyoruz?



Benim Bilgisayarlarım



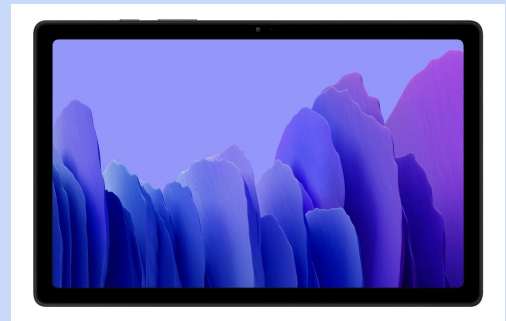
Thinkpad T480



Huawei P10

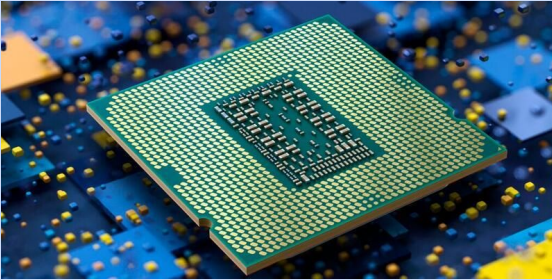
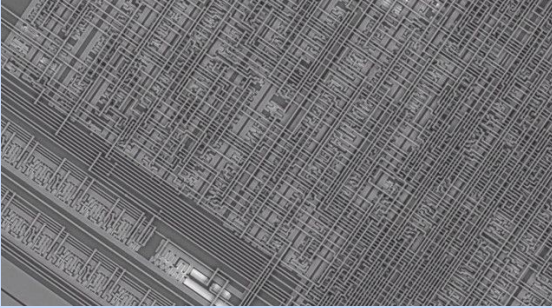


Raspberry Pi



Samsung Tablet

Bilgisayar



Sayı Sistemleri

Merhaba arkadaşlar!

01001101 01100101 01110010 01101000 01100001 01100010 01100001
00100000 01100001 01110010 01101011 01100001 01100100 01100001
10101111 01101100 01100001 01110010

10'luk Sistem

- 1
- 2
- 3
- ..
- 10
- 11
- 12

Örnek Hesaplamalar

$$10 + 10 = 20$$

$$01 + 11 = 12$$



2'lik Sistem - Binary

- 1
- 10
- 11
- 100
- 101
- 110
- 111
- 1000
-

Örnek Hesaplamalar

$$01 + 11 = 100$$

$$10 + 10 = 100$$

16'lık Sayı Sistemi

Binary gösterime göre daha kolay ifade edilebiliyor.

DECIMAL	HEX	BINARY
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

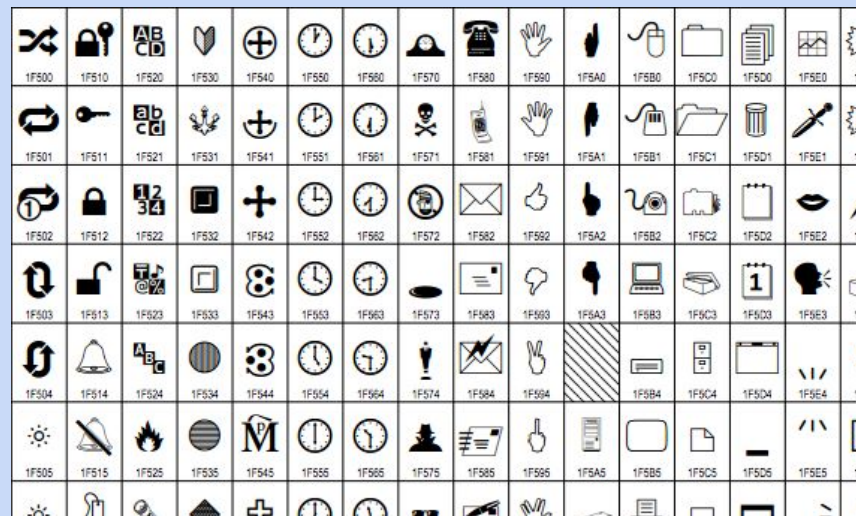
Yazıları Nasıl Sayılar İle İfade Ediyoruz ???

Merhaba arkadaşlar!

01001101 01100101 01110010 01101000 01100001 01100010 01100001
00100000 01100001 01110010 01101011 01100001 01100100 01100001
101011111 01101100 01100001 01110010

ASCII Tablosu & Unicodes

Hex	Dec	Char	Hex	Dec	Char	Hex	Dec	Char	Hex	Dec	Char
0x00	0	NULL null	0x20	32	Space	0x40	64	@	0x60	96	`
0x01	1	SOH Start of heading	0x21	33	!	0x41	65	A	0x61	97	a
0x02	2	STX Start of text	0x22	34	"	0x42	66	B	0x62	98	b
0x03	3	ETX End of text	0x23	35	#	0x43	67	C	0x63	99	c
0x04	4	EOT End of transmission	0x24	36	\$	0x44	68	D	0x64	100	d
0x05	5	ENQ Enquiry	0x25	37	%	0x45	69	E	0x65	101	e
0x06	6	ACK Acknowledge	0x26	38	&	0x46	70	F	0x66	102	f
0x07	7	BELL Bell	0x27	39	'	0x47	71	G	0x67	103	g
0x08	8	BS Backspace	0x28	40	(	0x48	72	H	0x68	104	h
0x09	9	TAB Horizontal tab	0x29	41	)	0x49	73	I	0x69	105	i
0x0A	10	LF New line	0x2A	42	*	0x4A	74	J	0x6A	106	j
0x0B	11	VT Vertical tab	0x2B	43	+	0x4B	75	K	0x6B	107	k
0x0C	12	FF Form Feed	0x2C	44	,	0x4C	76	L	0x6C	108	l
0x0D	13	CR Carriage return	0x2D	45	-	0x4D	77	M	0x6D	109	m
0x0E	14	SO Shift out	0x2E	46	.	0x4E	78	N	0x6E	110	n
0x0F	15	SI Shift in	0x2F	47	/	0x4F	79	O	0x6F	111	o
0x10	16	DLE Data link escape	0x30	48	0	0x50	80	P	0x70	112	p
0x11	17	DC1 Device control 1	0x31	49	1	0x51	81	Q	0x71	113	q
0x12	18	DC2 Device control 2	0x32	50	2	0x52	82	R	0x72	114	r
0x13	19	DC3 Device control 3	0x33	51	3	0x53	83	S	0x73	115	s
0x14	20	DC4 Device control 4	0x34	52	4	0x54	84	T	0x74	116	t
0x15	21	NAK Negative ack	0x35	53	5	0x55	85	U	0x75	117	u
0x16	22	SYN Synchronous idle	0x36	54	6	0x56	86	V	0x76	118	v
0x17	23	ETB End transmission block	0x37	55	7	0x57	87	W	0x77	119	w
0x18	24	CAN Cancel	0x38	56	8	0x58	88	X	0x78	120	x
0x19	25	EM End of medium	0x39	57	9	0x59	89	Y	0x79	121	y
0x1A	26	SUB Substitute	0x3A	58	:	0x5A	90	Z	0x7A	122	z
0x1B	27	FSC Escape	0x3B	59	;	0x5B	91	[	0x7B	123	{
0x1C	28	FS File separator	0x3C	60	<	0x5C	92	\	0x7C	124	
0x1D	29	GS Group separator	0x3D	61	=	0x5D	93	]	0x7D	125	}
0x1E	30	RS Record separator	0x3E	62	>	0x5E	94	^	0x7E	126	~
0x1F	31	US Unit separator	0x3F	63	?	0x5F	95	_	0x7F	127	DEL



Lojik



Bu sayıları kontrol ederek bir şeyler yapabilir miyiz?

Lojik

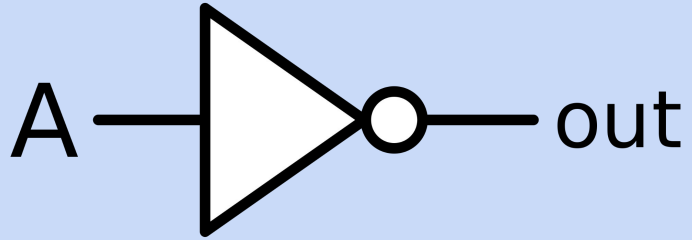
EVET

1

HAYIR

0

NOT (Değil) Kapısı

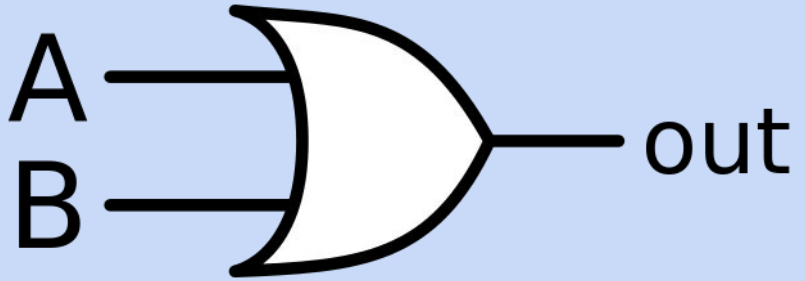


INPUT	OUTPUT
A	
0	0
1	1

INPUT	OUTPUT
A	
0	1
1	0

Doğruluk Tablosu (Truth table)

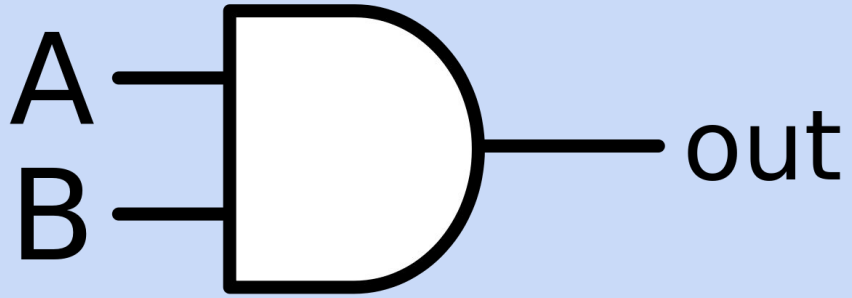
OR (Veya) Kapısı



INPUT		OUTPUT
A	B	
0	0	0
1	0	1
0	1	1
1	1	1

Doğruluk Tablosu (Truth table)

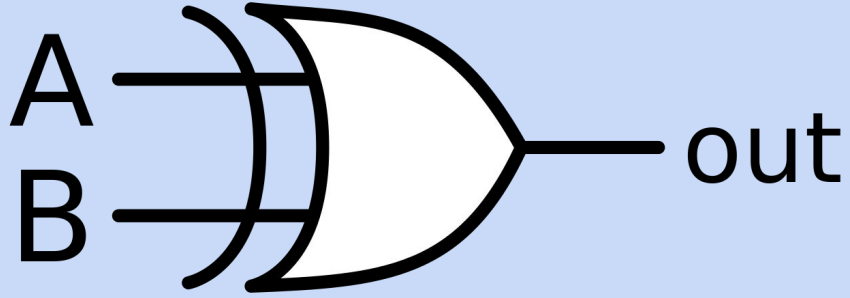
AND (Ve) Kapısı



INPUT		OUTPUT
A	B	
0	0	0
1	0	0
0	1	0
1	1	1

Doğruluk Tablosu (Truth table)

XOR Kapısı

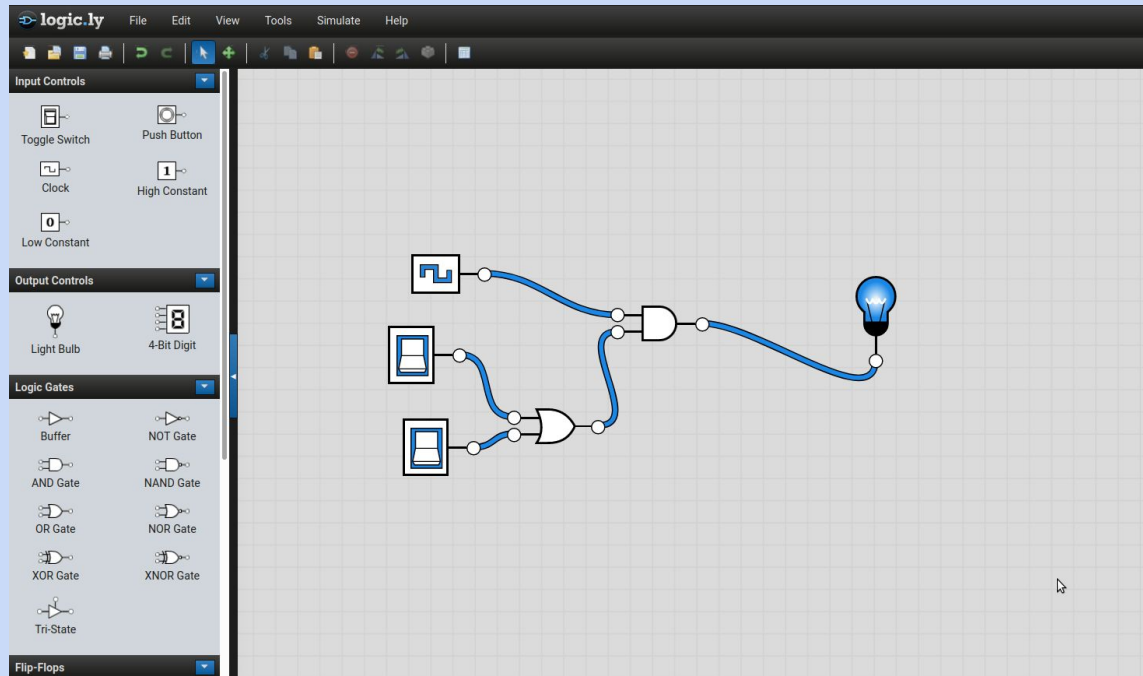


INPUT		OUTPUT
A	B	
0	0	0
1	0	1
0	1	1
1	1	0

Doğruluk Tablosu (Truth table)

Simülasyon

<https://logic.ly>



Gerçekten Bu Şekilde Bilgisayar Yapabilir Miyiz?



PEKİ BUNLARIN HEPSİNE BİR BİLGİSAYAR
PROGRAMINI OLUŞTURABİLMEK İÇİN İHTİYACIMIZ
OLACAK MI ?

SOYUTLAMA

İlk Bilgisayar Programımız

```
print("Merhaba Dünya")
```

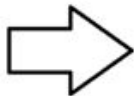
Bu Kadar Alt Seviyelere Nasıl Geçebiliyoruz

Compilers

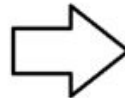
```
#include<stdio.h>

int main()
{
    printf("Hello, World!\n");
    return 0;
}
```

hello_world.c



Compiler



```
0110011000100010011000111
1100000001111111110000001
1111000110101010001100011
0011000100010011000111110
0000001111111110000001111
1000110101010001100011001
1000100010011000111110000
0001111111110000001111100
0110101010001100011001100
0100010011000111110000000
1111111110000001111100011
```

hello_world.o

Sorular



Kaynaklar

- <https://web.archive.org/web/20210702193243/http://www.cs.uah.edu/~rcoleman/Common/History/History.html>
- <https://www.livescience.com/20718-computer-history.html>
- https://tr.wikipedia.org/wiki/Turing_testi