مفاهیم و کاربردهای هوش مصنوعی در حسابداری

الگوریتم و تفکر الگوریتمی



بهرام صادقی بی غم

عضو هیات علمی دانشگاه و مشاورصنعت

و تجارت در حوزه هوش مصنوعی و هوش تجاری

وبسایت رسمی www.sadeghibigham.ir/

دو قلمرو متفاوت از مسائل در مواجهه با هوش مصنوعی

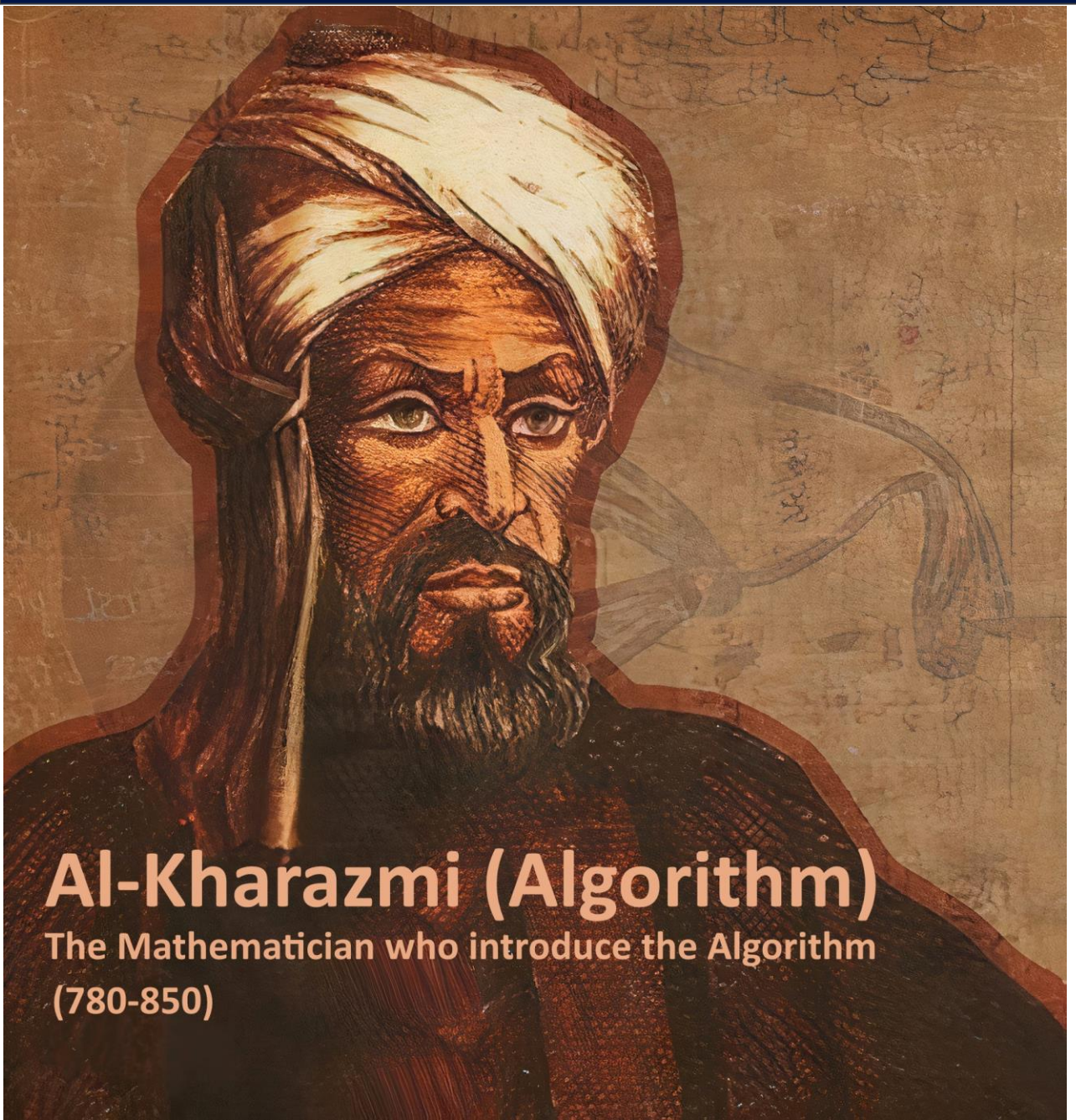
نوع مسئله	برای انسان	برای الگوریتم کلاسیک	برای یادگیری عمیق
ضرب اعداد بزرگ	سخت	بسیار آسان	غیرضروری
حل معادلات	سخت	بسیار آسان	ضعیف
مرتب‌سازی داده‌ها	سخت	بسیار آسان	ضعیف
تشخیص چهره	آسان	بسیار سخت	بسیار خوب
خواندن دست‌خط	آسان	بسیار سخت	بسیار خوب
درک گفتار	آسان	بسیار سخت	بسیار خوب
تشخیص اشیاء در تصویر	آسان	بسیار سخت	بسیار خوب

یاداوری: سه اصل موفقیت در استفاده از ابزارهای هوش مصنوعی

- اصل اول: رویکرد الگوریتمی
- اصل دوم: پذیرش کامل مسئولیت
- اصل سوم: پرامپت نویسی صحیح

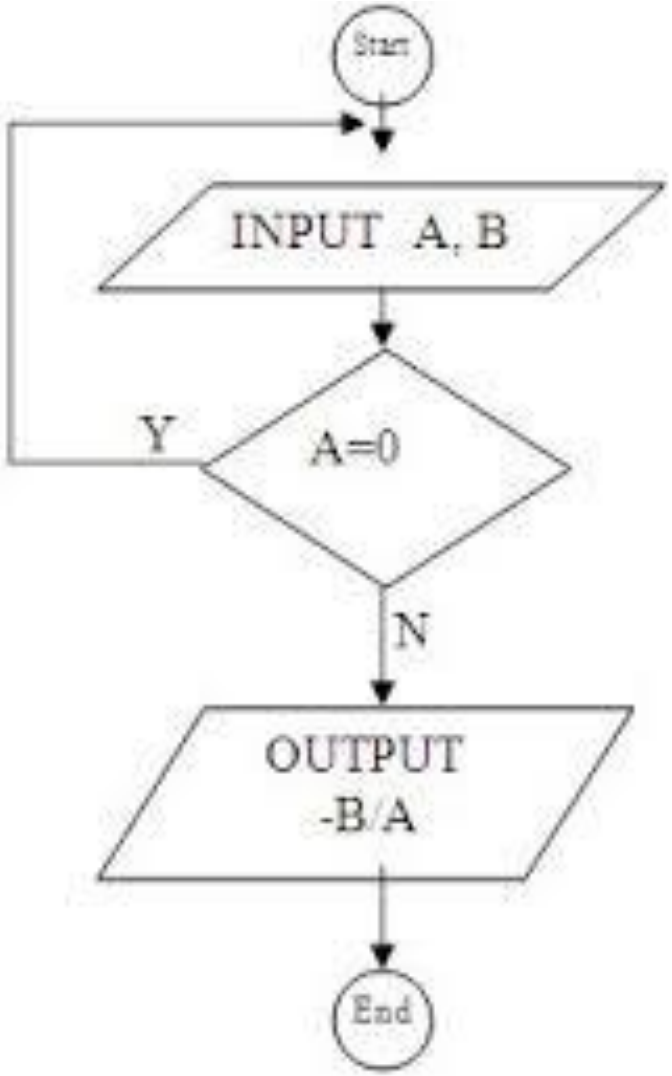


خوارزمی



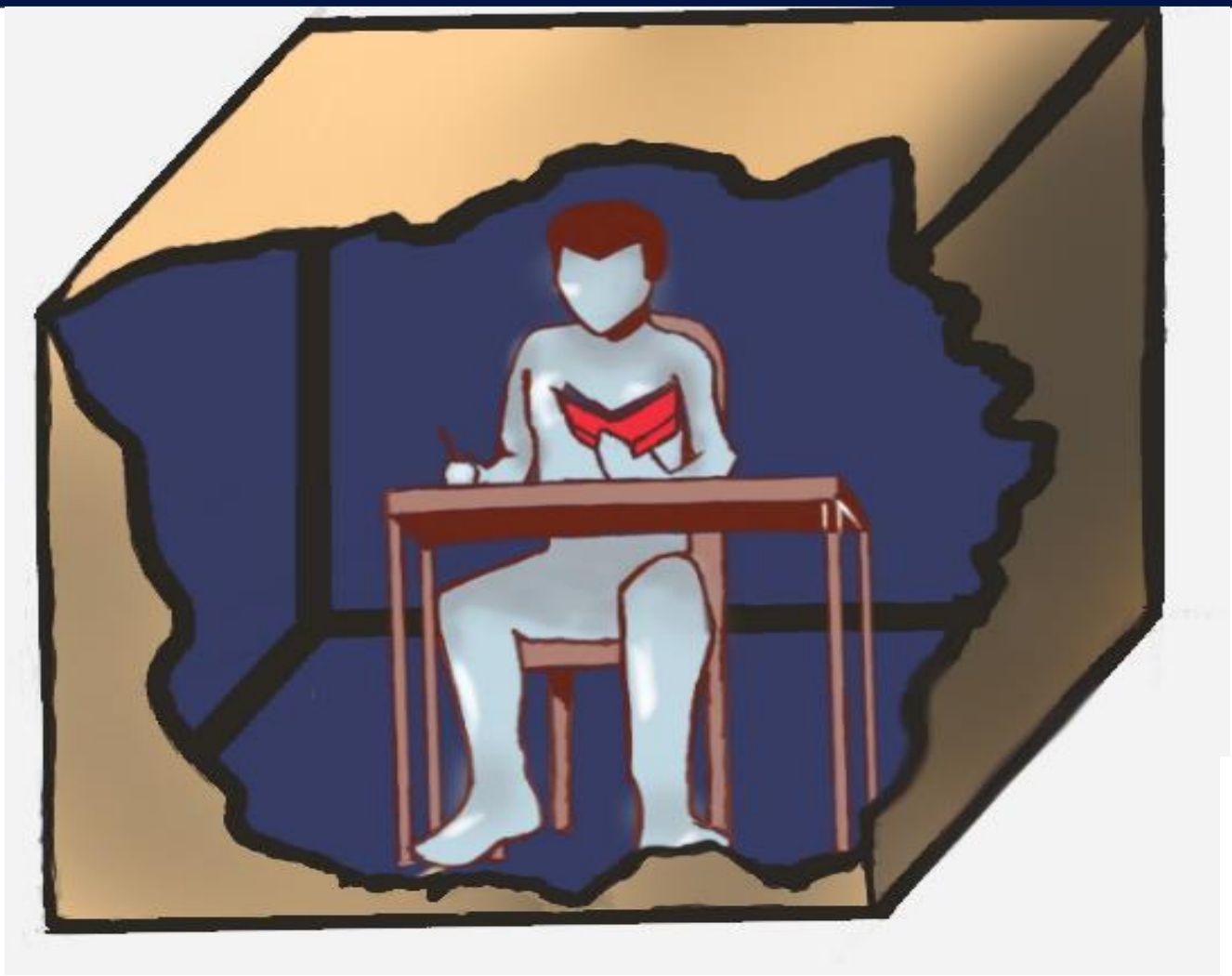
محمدابن‌موسی‌الخوارزمی دانشمند ایرانی در قرون وسطی (بین سال‌های ۷۸۰-۸۵۰ میلادی) و اهل خوارزم (بخش جنوبی ازبکستان کنونی) بود. خوارزمی ریاضی‌دان، منجم و به قول برخی تاریخ‌نویسان، همه‌چیزدان زمان خود بوده است و کتاب جبر و مقابله‌ی او هنوز در برخی محافل علمی مورد استفاده است. او در کتاب خود و در مورد معادلات از عبارت شیئ برای نشان دادن مجهول استفاده می‌کرد. به‌عنوان نمونه، در کتاب او، عبارت "مالی به اضافه‌ی شیئ‌هایی، مساوی عددی است" به معنای معادله‌ی $x^2 + ax = b$ است. وقتی که اروپائیان در اسپانیا کتاب‌های او را به زبان خود ترجمه می‌کردند، کلمه "شیء" را به صورت "Xei" نوشتند و به مرور از قرن ۱۶ به بعد و وقتی که معادلات ریاضی به صورت نمادگذاری‌شده نوشته شد، اروپایی‌ها حرف اول آن یعنی «X» را به جای مجهول استفاده کردند که امروزه، استفاده از آن به جای متغیر و یا مجهول در ریاضی هم‌چنان مرسوم است.

الگوریتم



در قرن نوزدهم، کلمه‌ی الگوریثموس در زبان فرانسوی به الگوریزم تغییر شکل پیدا کرد و پس از مدتی به شکل الگوریتم وارد زبان انگلیسی شد. در اواخر قرن ۱۹ میلادی، استفاده این کلمه فراگیر شد و در محافل علمی به هر مجموعه دستورالعمل ساده که برای انجام یک کار مورد استفاده قرارگیرند الگوریتم گفته شد. در حال حاضر الگوریتم‌ها را گاها به صورت شبه کدو یا با استفاده از فلوچارت نشان می‌دهند.

سطح هوش کامپیوتر؟



کامپیوتر، مانند یک جعبه در نظر گرفته می شود که درون آن شخصی حرف‌گوش‌کن، سریع و با قدرت ذهنی محدود نشسته و دقیقا طبق دستورالعمل‌ها (الگوریتم)، گام به گام و با سرعت عمل می‌کند.

زبانهای سطح پایین (کامپیوتر بسیار کند ذهن است و همه چیز را باید ریز به ریز و با ترتیب درست به او گفت. مانند زبان اسمبلی

زبانهای سطح بالا (کامپیوتر کمی فهمیده تر است و زبان برنامه نویسی به زبان انسان نزدیک تر است. با این حال مراحل حل مساله را باید ریز به ریز و با ترتیب درست به او گفت. مانند زبان پایتون

زبان صحبت با هوش مصنوعی (کاملا به زبان انسان با کامپیوتر صحبت می شود اما نیاز به بیان الگوریتمی همچنان ضروری است (رویکرد نوروسیمبلیک) وگرنه کامپیوتر نمی تواند از پس کارهای تخصصی بر بیاید.

دید الگوریتمی

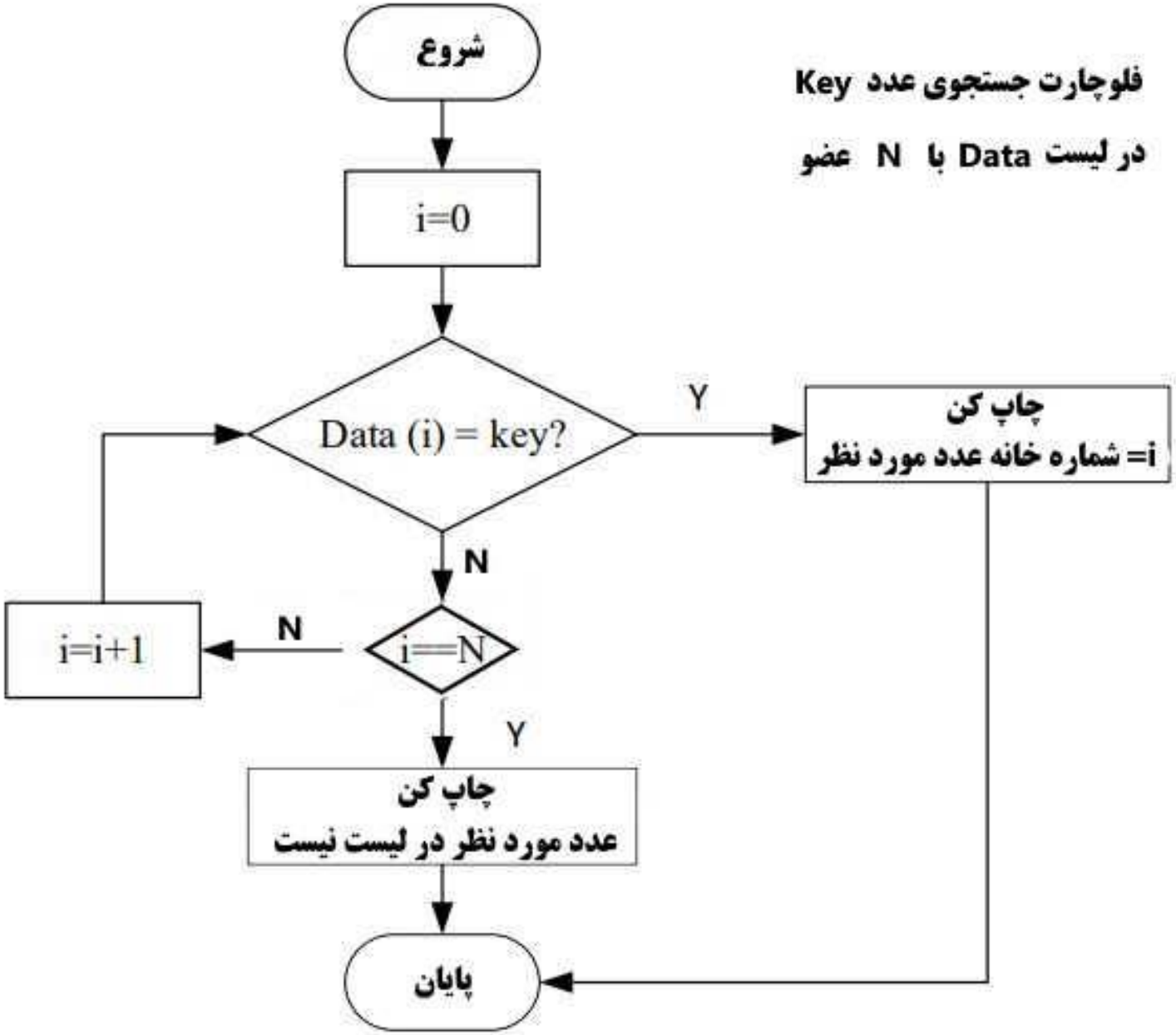
در نگاه الگوریتمی باید فرض کرد که کامپیوتر وسعت دید ندارد

انسان چگونه از بین ۷ عدد داده شده، کمترین عدد را انتخاب می کند؟

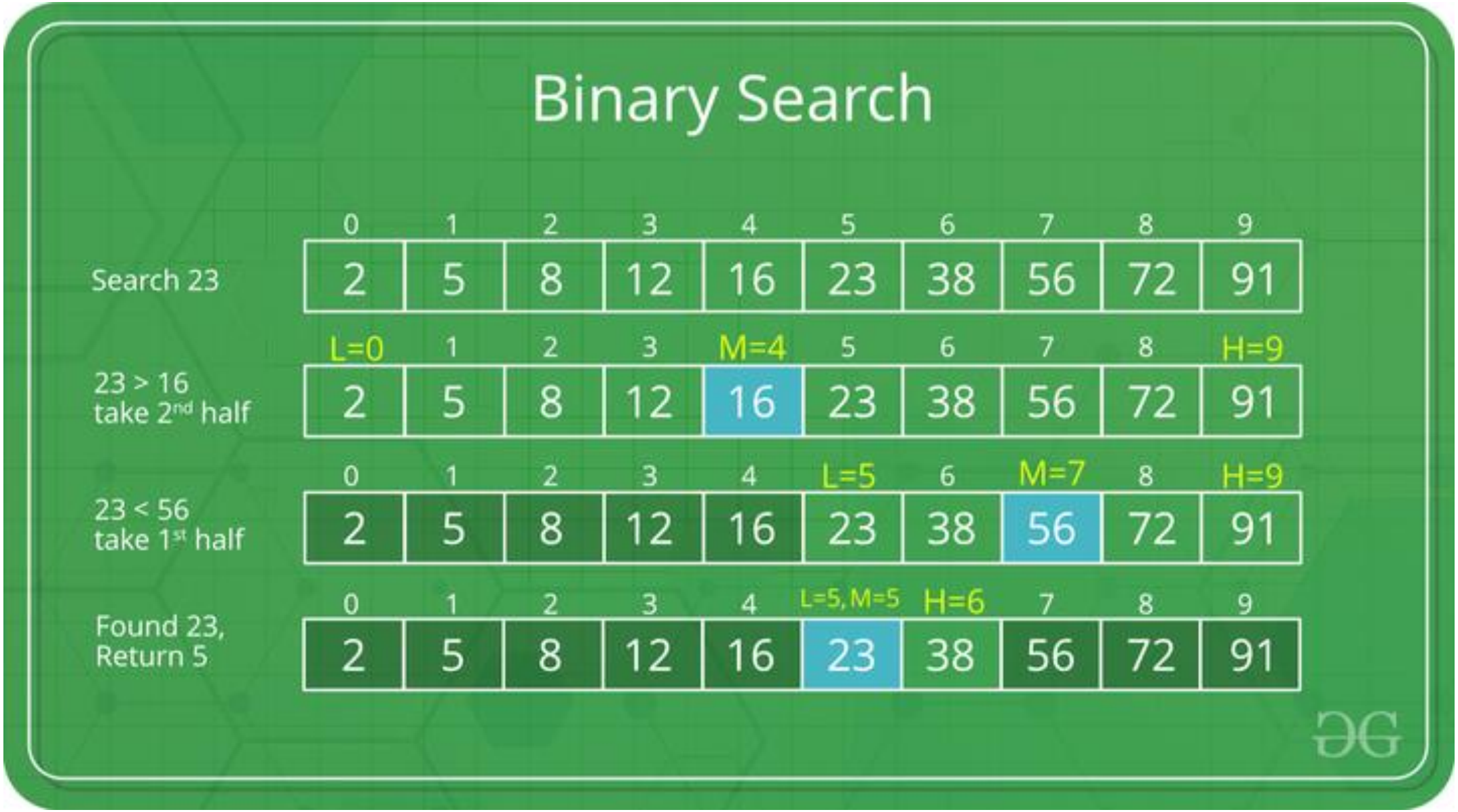
8, 5, 11, 6, 2, 14, 4

کامپیوتر چگونه باید این کار را بکند؟ آیا شیوه کار یکسان است؟
یک عدد دلخواه مانند ۹ را چگونه در این لیست جستجو می کند؟
اگر بخواهیم لیست را بازنویسی کند و به جای همه اعداد تک رقمی مقدار ۰ بنویسد الگوریتم باید چگونه باشد؟
چگونه باید از کامپیوتر بخواهیم که این اعداد را به ترتیب برعکس بنویسد؟
الگوریتمی بنویسید که این اعداد را به صورت صعودی مرتب کند.

فلوچارت جستجوی خطی یا ترتیبی

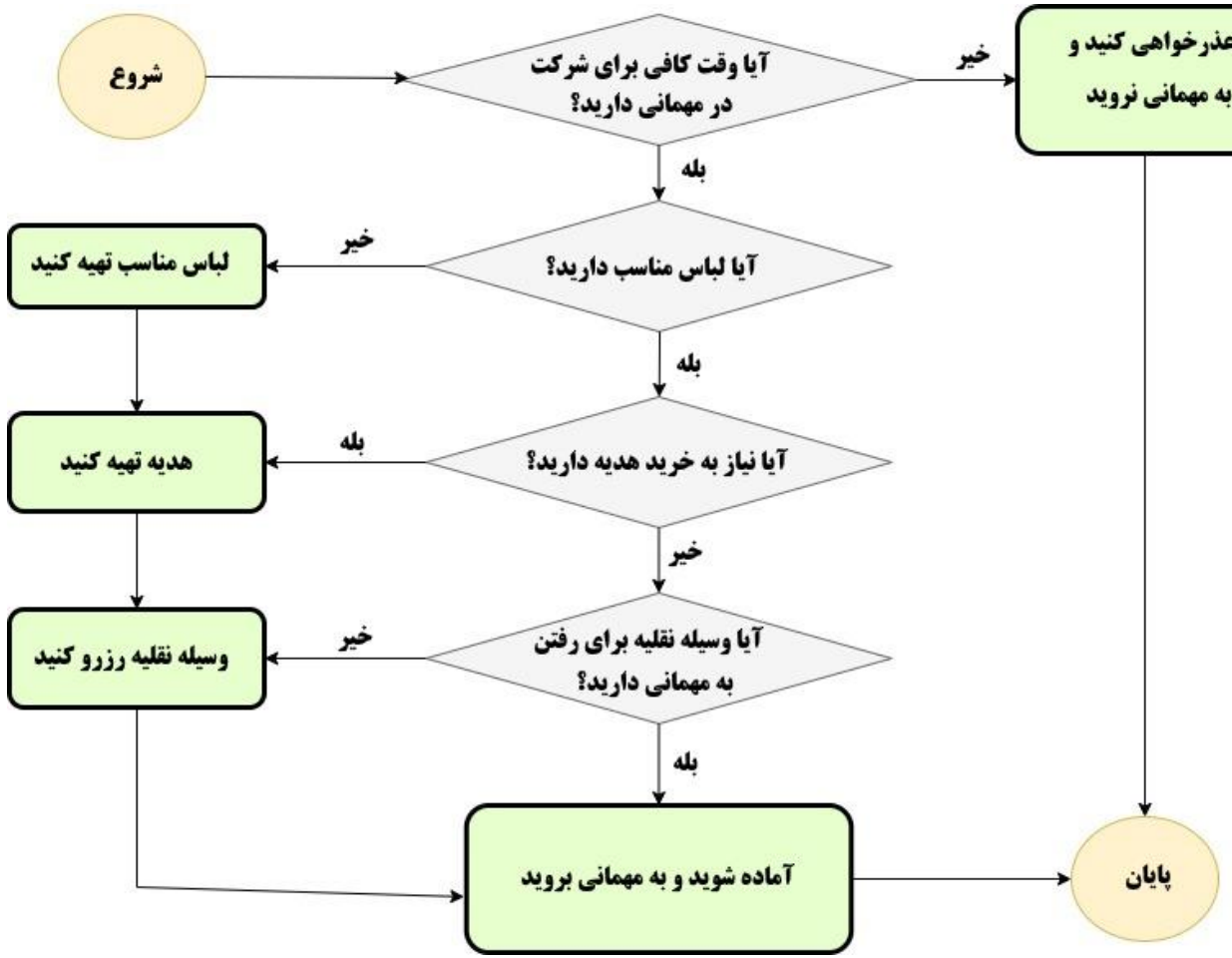


جستجوی دودویی در آرایه مرتب



دستورات پایه ای برای الگوریتم

- انتساب (دادن مقدار به متغیر)
- ورودی/ خروجی (مقدار نهایی متغیر را چاپ کن)
- شرط و تصمیم گیری
- تکرار و انواع حلقه ها
- توقف و بازگشت (پایان منطقی الگوریتم)



آیا این الگوریتم ایراد دارد؟

روشهای بیان الگوریتم

- توضیح متنی و مرحله ای
- فلوچارت
- شبه کد

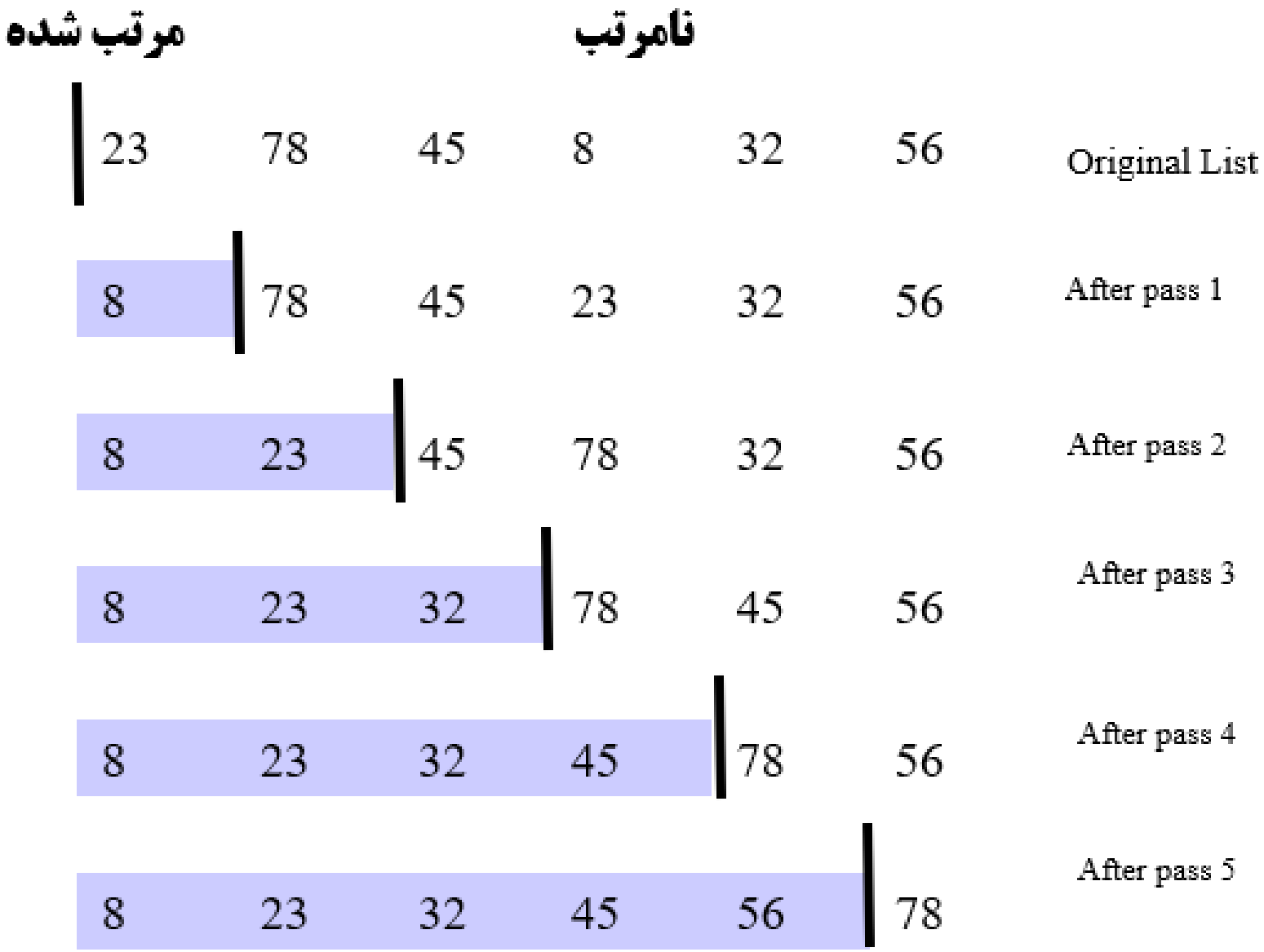
فلوچارت

- در رسم فلوچارت هدف این است که جریان کنترل کاملاً یکتا، قابل دنبال‌کردن و بدون تفسیر شخصی باشد.
- معیار طلایی تشخیص فلوچارت صحیح یک نفر بدون توضیح شفاهی بتواند فقط با دنبال‌کردن فلش‌ها برنامه را بنویسد.

ایرادهای رایج در رسم فلوچارت

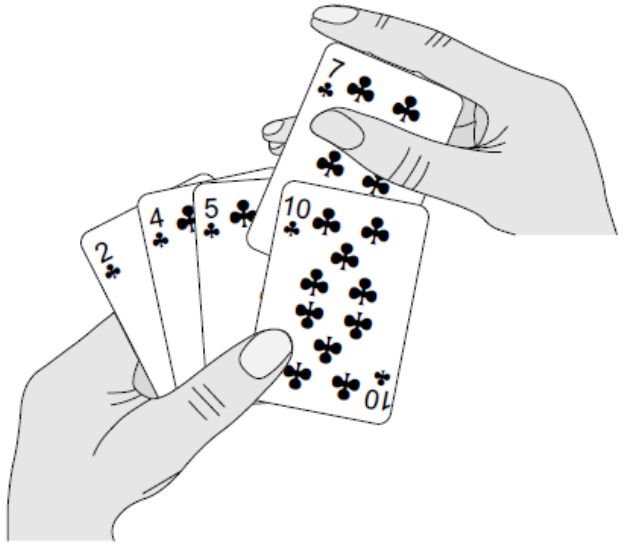
- چند خروجی از یک باکس پردازش
- نداشتن برچسب روی خروجی شرط (لوزی)
- خروجی های بیشتر از یک شرط (لوزی فقط دو حالت خروجی (شبیه بله و خیر) دارد)
- نداشتن پایان منطقی برای مسیرها
- حلقه‌های نامشخص (شرط نامفهوم)
- استفاده از جملات مبهم (مانند بررسی کن یا پردازش کن)

مرتب سازی انتخابی



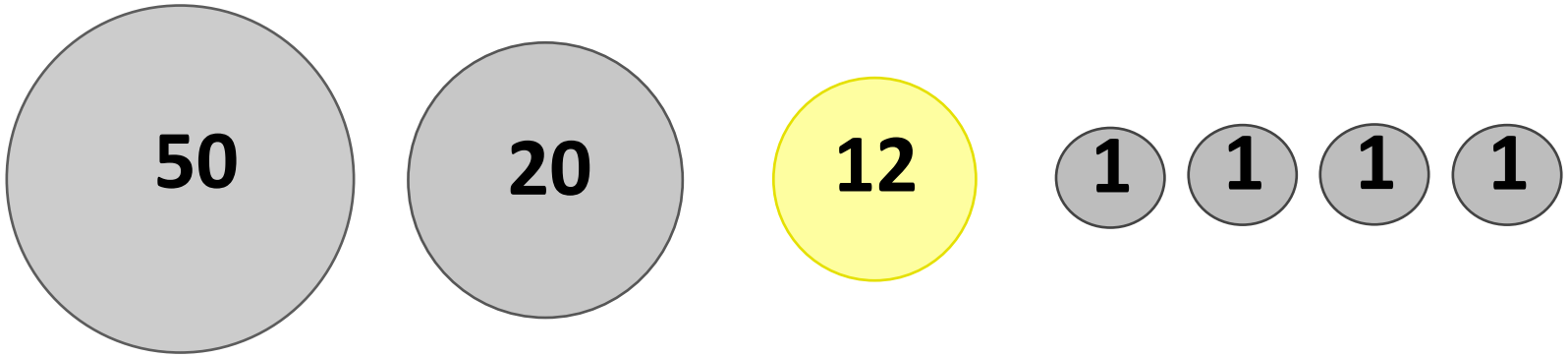
مرتب سازی درجی

مرتب شده			نامرتب			
23	78	45	8	32	56	Original List
23	78	45	8	32	56	After pass 1
23	45	78	8	32	56	After pass 2
8	23	45	78	32	56	After pass 3
8	23	32	45	78	56	After pass 4
8	23	32	45	56	78	After pass 5



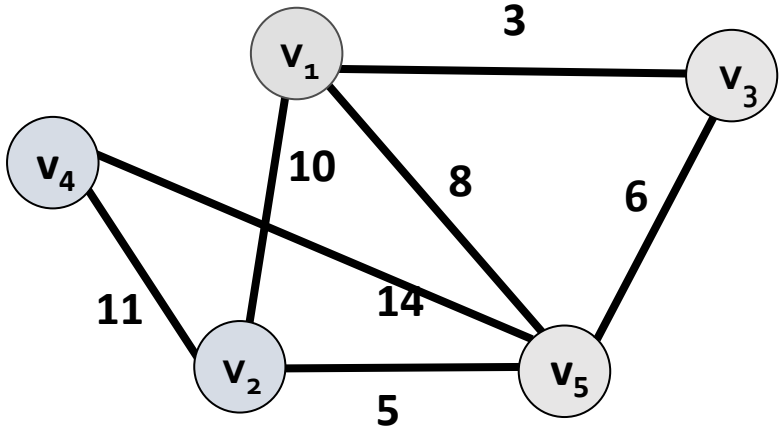
یک مساله سخت: مساله پرداخت کمترین تعداد سکه

- Given a value $86, if we want to make a change for $86, and we have an infinite supply of { 1, 5, 10, 12, 20, 50, 100} valued coins, what is the minimum number of coins needed to make the change?



آیا این روش بهینه است؟

معرفی تعدادی مساله معروف، ساده و سخت



- درخت پوشای فراگیر
- کوتاهترین مسیر
- فروشنده دوره گرد
- دور همیلتونی
- کدگذاری هافمن
- مساله کوله پشتی

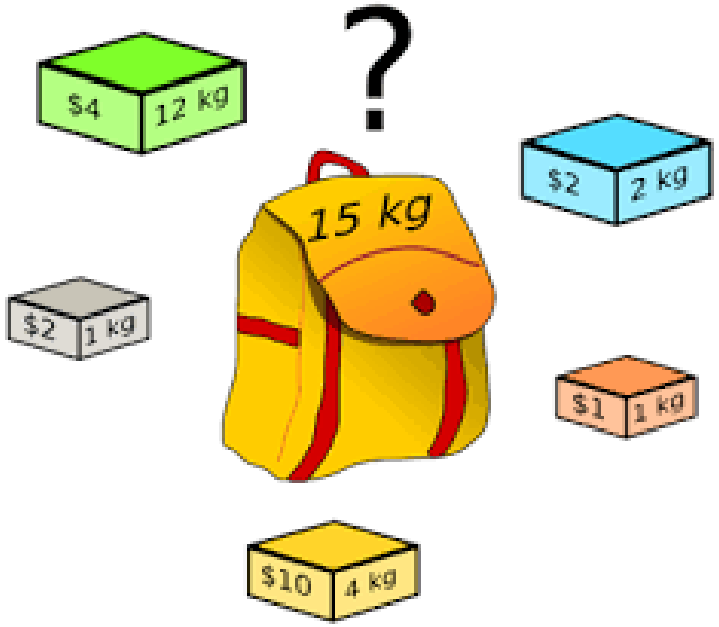
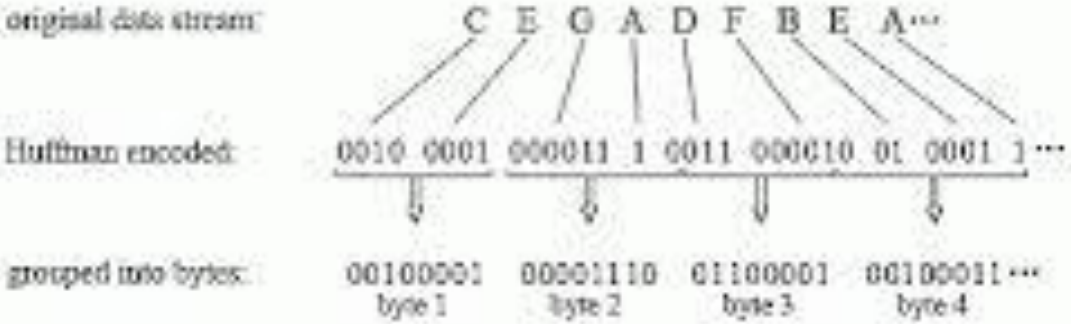


FIGURE 27-3
Huffman encoding. The encoding table assigns each of the seven letters used in this example a variable length binary code, based on its probability of occurrence. The original data stream composed of these 7 characters is translated by this table into the Huffman encoded data. Since each of the Huffman codes is a different length, the binary data need to be regrouped into standard 8 bit bytes for storage and transmission.

Example Encoding Table

letter	probability	Huffman code
A	.154	1
B	.110	01
C	.072	0010
D	.063	0011
E	.059	0001
F	.015	000010
G	.011	000011



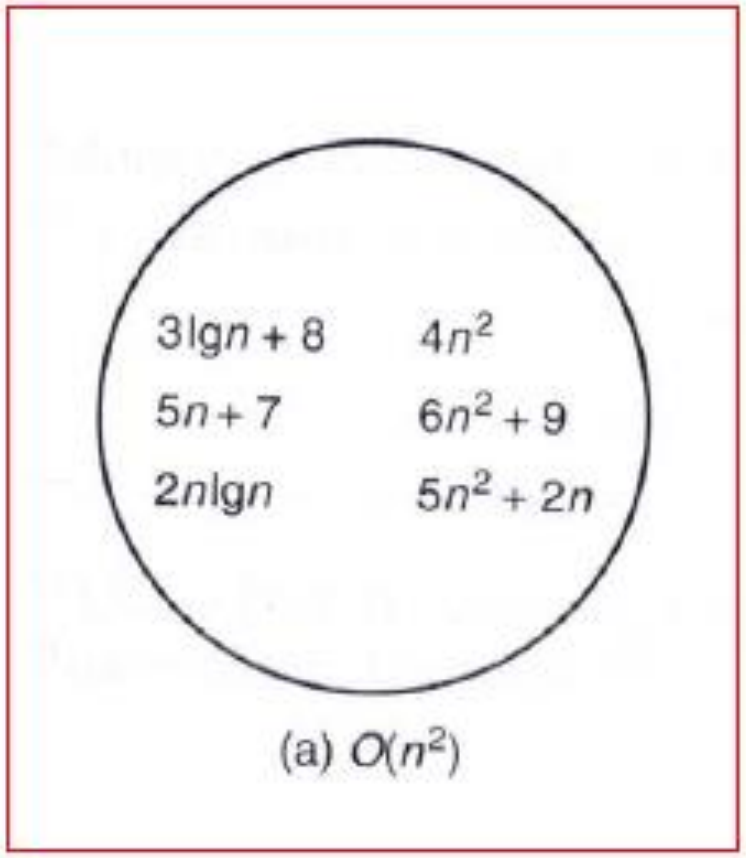
سرعت اجرای الگوریتم و فضای مصرفی

- فرض کنید **n** تعداد داده های ورودی به یک الگوریتم باشد (مانند تعداد اعداد، یا تعداد گره های یک گراف، یا تعداد پیکسل های یک عکس، ...)
- تعداد تقریبی عملیاتی که الگوریتم شما برای رسیدن به پاسخ انجام می دهد بسیار مهم است.
- این تعداد را به سادگی نمی توان شمرد اما یک برآورد تقریبی و یا مرتبه (**Order**) آن را می توان محاسبه کرد.
- درخت پوشای دو مورد بسیار مهم است. مرتبه زمانی و مرتبه فضایی.
- مرتبه زمانی بسیار بسیار مهم تر است.
- فرض کنید هر عمل اصلی در کامپیوتر در زمان یک نانو ثانیه انجام شود (10^{-9})

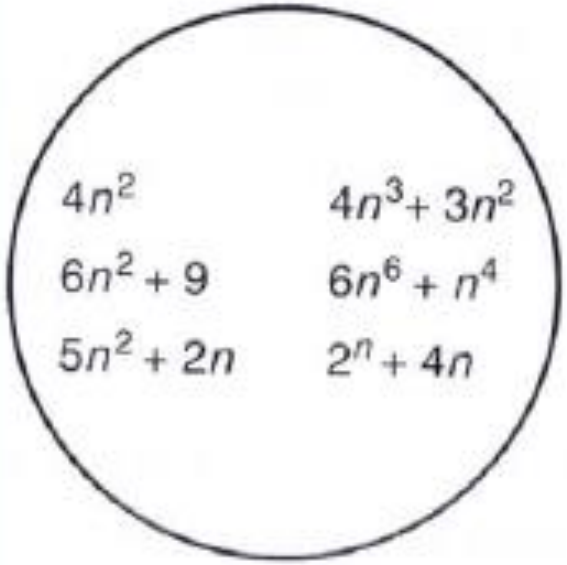
زمان اجرا بسیار حیاتی است.

n	$f(n) = lg\, n$	$f(n) = n$	$f(n) = n\, lg\, n$	$f(n) = n^2$	$f(n) = n^3$	$f(n) = 2^n$
10	0.003 μs*	0.01 μs	0.033 μs	0.10 μs	1.0 μs	1 μs
20	0.004 μs	0.02 μs	0.086 μs	0.40 μs	8.0 μs	1 ms†
30	0.005 μs	0.03 μs	0.147 μs	0.90 μs	27.0 μs	1 s
40	0.005 μs	0.04 μs	0.213 μs	1.60 μs	64.0 μs	18.3 min
50	0.006 μs	0.05 μs	0.282 μs	2.50 μs	125.0 μs	13 days
10^2	0.007 μs	0.10 μs	0.664 μs	10.00 μs	1.0 ms	4×10^{13} years
10^3	0.010 μs	1.00 μs	9.966 μs	1.00 ms	1.0 s	
10^4	0.013 μs	10.00 μs	130.000 μs	100.00 ms	16.7 min	
10^5	0.017 μs	0.10 ms	1.670 ms	10.00 s	11.6 days	
10^6	0.020 μs	1.00 ms	19.930 ms	16.70 min	31.7 years	
10^7	0.023 μs	0.01 s	2.660 s	1.16 days	31,709 years	
10^8	0.027 μs	0.10 s	2.660 s	115.70 days	3.17×10^7 years	
10^9	0.030 μs	1.00 s	29.900 s	31.70 years		

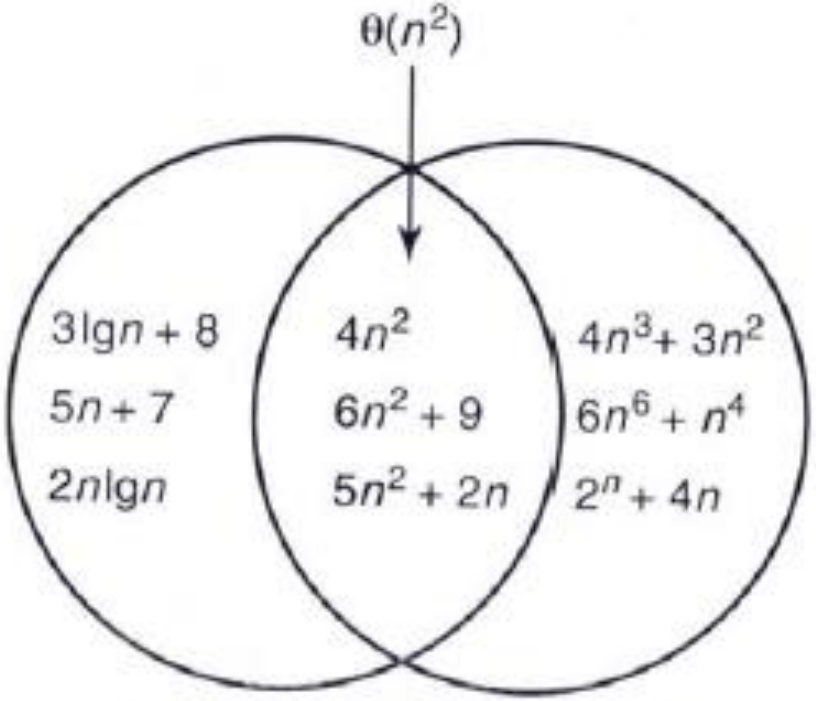
سه نماد معروف – معرفی در حد نماد



(a) $O(n^2)$



(b) $\Omega(n^2)$



(c) $\theta(n^2) = O(n^2) \cap \Omega(n^2)$

نگاه الگوریتمی در کار با هوش مصنوعی

- هوش مصنوعی باهوش است اماعقل کل نیست
- خطاهای زیادی دارد و همواره خواهد داشت
- برای این موجود با هوش محدود باید دستورات را الگوریتمی داد
- برای هر کارمندتان باید به نسبت هوش و توانایی او کارها را جزئی و مرحله به مرحله داد
- مسوولیت کار کارمندانتان به طور کامل متوجه شماست.



Any question?