



# دوره آموزشی طراحی و توسعه وب

Web Design and Development Training Course



جلسه نهم | Ninth Session

## تفکر رایانشی؛ رویکردی برای حل مسئله

Computational Thinking; An Approach to Problem Solving

پوریا الوانی

Pourya Alvani

توسعه دهنده نرم افزار در مرکز مطالعات گروه نرم افزاری پارت  
Software Developer at Part Software Group Research Institute

# در مورد چی صحبت می‌کنیم؟

What are we talking about?



## آشنایی با تفکر رایانشی

Familiarity with Computational Thinking

۱

## ماده خام

Raw Material

۲

## بازنمایی اطلاعات توسط کامپیوتر

Representation of Information by Computer

۳



# در مورد چی صحبت می‌کنیم؟

What are we talking about?



## الگوریتم‌های جستجو

Search Algorithms

۴

## مسیر پوشای کمینه

Minimum Path

۵



# 01.



## آشنایی با تفکر رایانشی

Familiarity with Computational Thinking



## تفکر رایانشی چیست؟

❖ به فرآیندهای فکری موبوط به فرمول بندی مسائل که بتوان راه حل آنها را به صورت مراحل و الگوریتم‌های محاسباتی نشان داد ، تفکر رایانشی می گویند.

### ❖ فرآیند تفکر رایانشی

- تجزیه: نیاز است مسائل پیچیده به مسائل ساده تر تقسیم شوند
- پیرایش: از داده های غیرضروری صرف نظر کنید
- الگویابی: یافتن الگوهای مشترک در مسئله‌ها
- الگوریتم: طراحی زیرمسئله های کوچکتر برای رسیدن به راه حل مسئله اصلی
- ارزیابی: تلاش برای بهبود راه‌حل ارائه شده



# اهمیت تفکر رایانشی

❖ یادگیری فرآیند تعریف راه حل های یک مسئله

❖ یادگیری راه حل هایی مبتنی بر کامپیوتر

❖ بهبود توانایی حل مسائل پیچیده کامپیوتری



# 02.

## ماده خام

Raw Material







# عددهای دودویی

## ❖ عدد دودویی چیست؟

○ به هر عددی که از صفر یا یک تشکیل شده باشد عدد دودویی یا مبنا دو می گویند.

○ مثال : ۱۰۰ ، ۱۰۱۰۱ ، ۱۰۰۰۱

## ❖ کار با سیستم اعداد دودویی

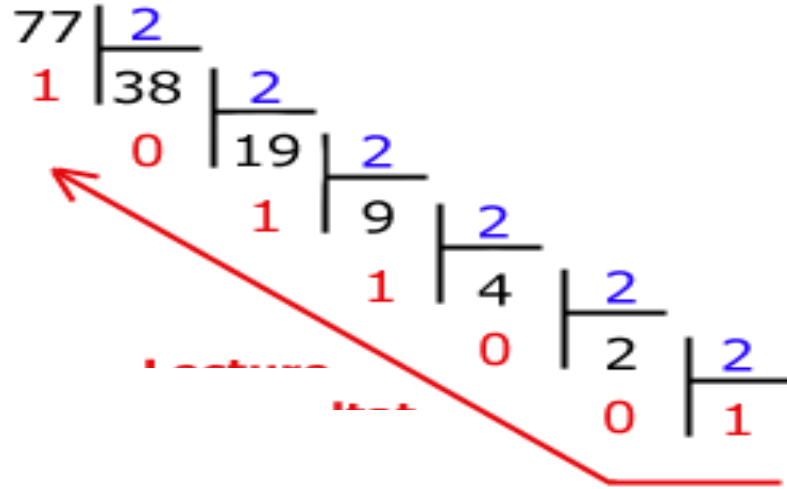
1	1	0	0	1	0	1	0
↓	↓	↓	↓	↓	↓	↓	↓
$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
↓	↓	↓	↓	↓	↓	↓	↓
128	64	0	0	8	0	2	0

$$11001010 = 128 + 64 + 8 + 2 = 202$$

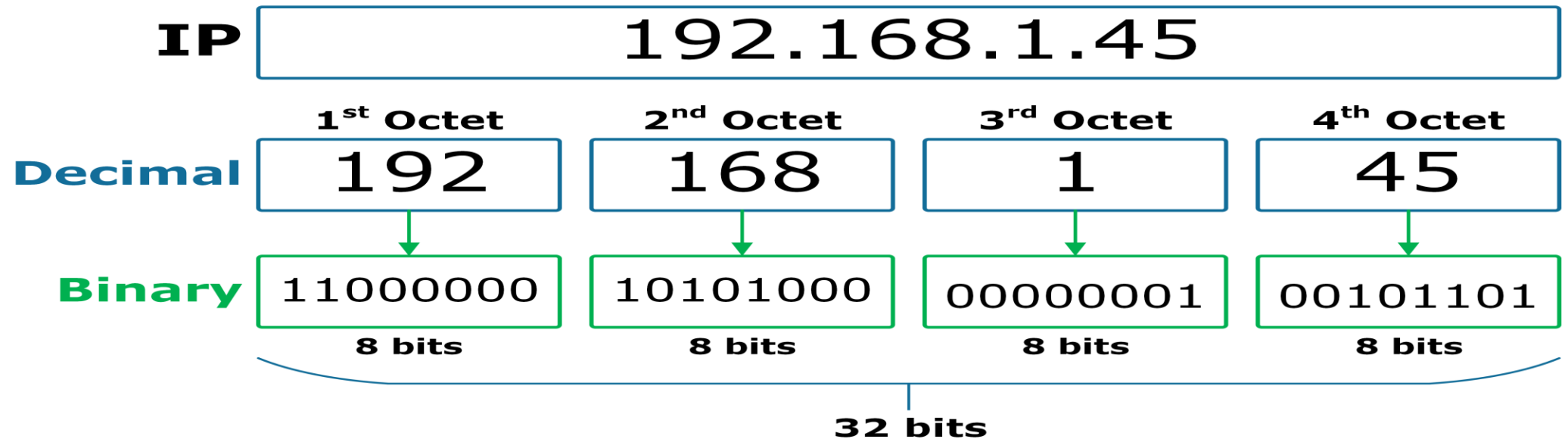




# تبدیل عدد دهدهی به دودویی



$$(77) = 1001101$$





# اطلاعات و داده

## ❖ فرق اطلاعات با داده

- داده همان اطلاعاتی است که به صورت اعداد باینری درآمده است تا برای کامپیوتر خوانا باشد.
- داده ها خام و بدون ساختار هستند و اما اطلاعات زمینه و بینشی را همراه دارد.

## ❖ تبدیل اطلاعات به داده

- انواع اطلاعات : عکس ، فیلم ، صدا ، متن و ...
- هر اطلاعاتی با توجه به نوع آن توسط الگوریتمی به اعداد باینری تبدیل می شود.
- کمی جلوتر نمونه از الگوریتم تبدیل متن به باینری را خواهید دید.



## اطلاعات و داده

### ❖ تبدیل متن به داده

- به ترتیب برای هر حرف الفبا یک عدد را در نظر بگیرید.
- حال هر کدام از این اعداد را به صورت باینری تبدیل کنید.
- سپس کافی است به جای هر حرف الفبا معادل باینری عددی اختصاص دادین را قرار دهید.

1	2	3	4	5	6	7	8	9	10	11	12	13
A	B	C	D	E	F	G	H	I	J	K	L	M
14	15	16	17	18	19	20	21	22	23	24	25	26
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

**CANGROW : 00011 00001 01110 00111 10010 10001 10111**

### ❖ چالش‌ها

- آیا همیشه از حروف الفبا استفاده می‌شود؟
- چگونه با کمترین بیت بتوانیم هر متنی را به باینری تبدیل کنیم؟

# 03.

## بازنمایی اطلاعات توسط کامپیوتر

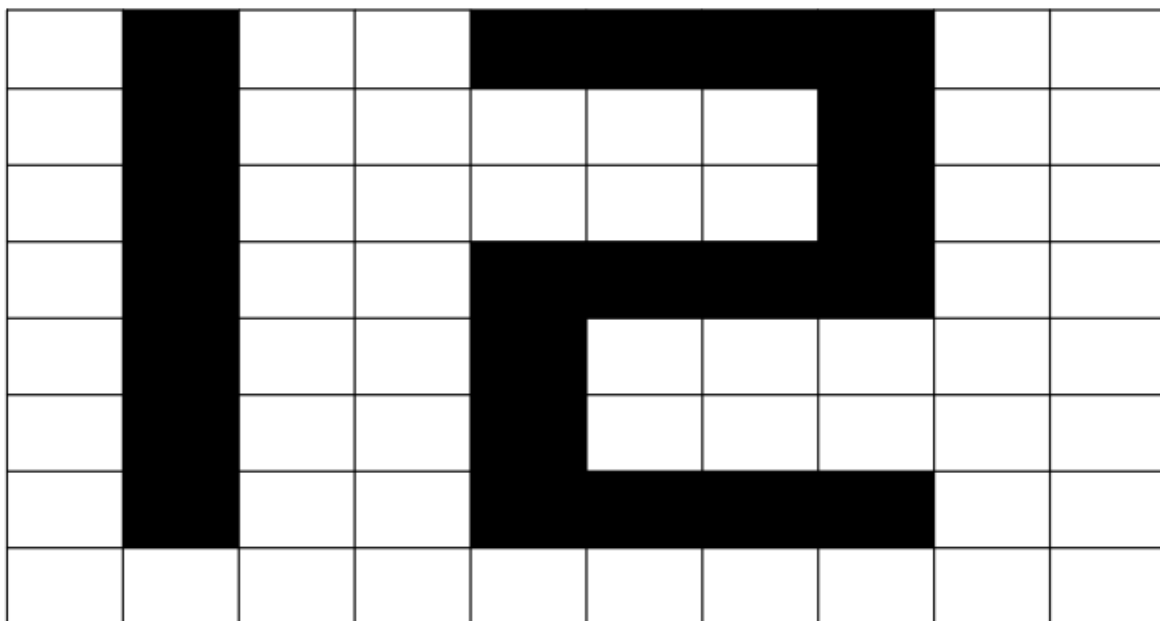
Representation of Information by Computer





## بازنمایی تصاویر

❖ هر خانه مشکی را ۱ و هر خانه سفید را ۰ در نظر بگیرید.



1 1 2 4 2  
1 1 5 1 2  
1 1 5 1 2  
1 1 2 4 2  
1 1 2 1 5  
1 1 2 1 5  
1 1 2 4 2  
10

❖ با توجه به تصویر و اعداد چه الگویی را پیدا کردید؟

❖ آیا می شود برای تصاویر رنگی هم این الگو را پیاده کرد؟



## فشرده سازی متن

❖ رایانه‌ها داده‌های خیلی زیادی را ذخیره و مبادله می‌کنند، پس مجبورند از کم‌ترین فضای ذخیره‌سازی استفاده کنند یا برای انتقال اطلاعات در یک اتصال شبکه‌ای، کمترین زمان را صرف کنند. بنابراین متن‌ها را مشاه شیوه زیر فشرده می‌کنند.

### ❖ فشرده‌سازی:

• پیدا کردن الگویی بین حروف و عبارات

Pitter patter



pitter patter



## فشرده سازی متن

❖ متن زیر را در نظر بگیرید می توان آن را به صورت زیر فشرده کرد.

Pease porridge hot ,  
Pease porridge cold ,  
Pease porridge in the pot ,  
Nine days old.  
Some like it hot,  
Some like it cold,  
Some like it in the pot,  
Nine days old,



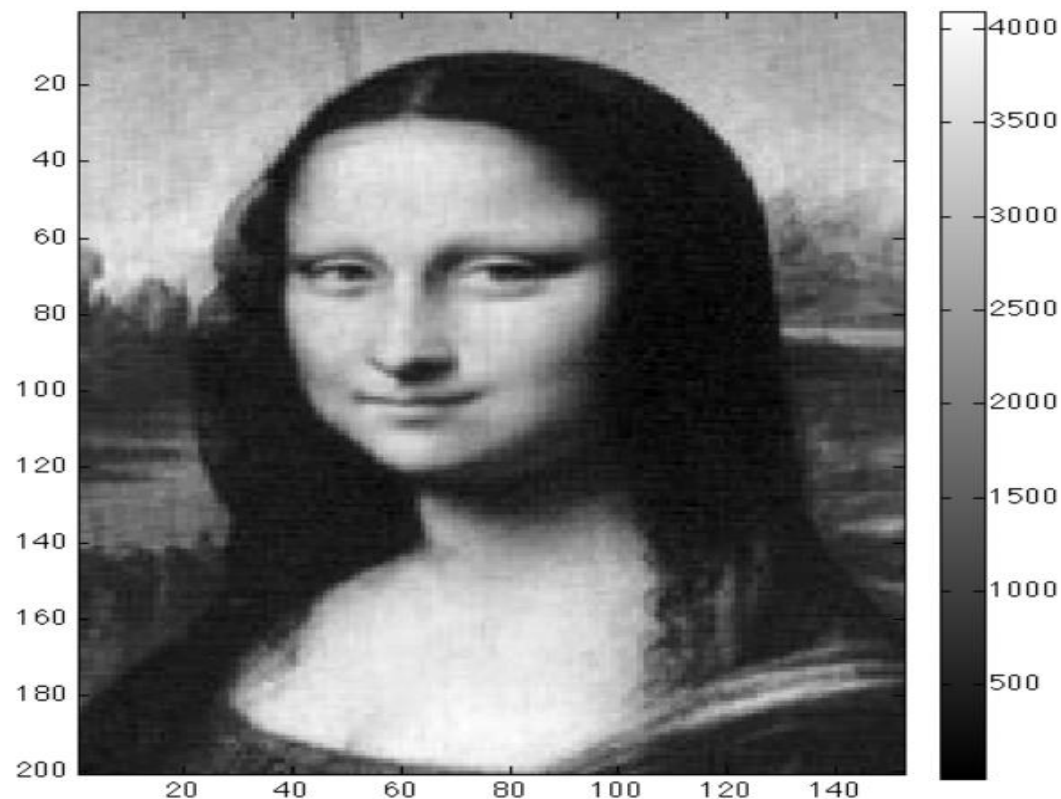
Pease porridge hot ,  
Pease porridge cold ,  
Pease porridge in the pot ,  
Nine days old,  
Some like it hot,  
Some like it cold,  
Some like it in the pot,  
Nine days old,





## تشخیص و تصحیح خطا در داده

❖ معمولا ما گمان می‌کنیم که وقتی داده‌های روی دیسک ذخیره می‌شوند یا از یک رایانه به رایانه دیگری منتقل می‌شوند، به هیچ وجه تغییر نمی‌کنند. اما گاهی اشتباهی رخ می‌دهد و داده‌ها به‌طور تصادفی تغییر می‌کنند.





## تشخیص و تصحیح خطا در داده

❖ می‌دانیم که داده‌ها در کامپیوتر به صورت صفر و یک است ، با توجه به این برای تشخیص و تصحیح خطا می‌توانیم از تکنیک زیر استفاده کنیم :

0	1	0	1	1	0	1	0	0	0	0	← Parity even bit
1	1	1	0	0	1	1	1	0	1	1	
0	0	1	0	0	0	1	1	0	0	1	
1	1	0	1	1	0	1	0	1	1	1	
0	1	0	0	0	1	0	0	1	0		



# 04.

## الگوریتم‌های جستجوی

Search algorithms



## جستجوی

❖ اغلب رایانه باید اطلاعاتی را از میان مجموعه بزرگی از داده‌ها پیدا کنند. این کار به روشی نیاز دارد که هم سریع باشد و هم کارای لازم را داشته باشد.

### ❖ انواع روش‌های جستجو :

1. جستجوی خطی
2. جستجوی دودویی
3. جدول درهم سازی




## جستجوی خطی

❖ جستجوی خطی که اغلب به عنوان جستجوی متوالی شناخته می‌شود ، اساسی ترین تکنیک جستجوی است. در این نوع جستجوی ، کل فهرست را مرور می‌کنید و سعی می‌کنید برای یک عنصر تطابق را پیدا کنید . اگر مطابقت پیدا کردید، آدرس عنصر هدف برگردانه می‌شود

❖ آرایه زیر را در نظر بگیرید ، می‌خواهیم عنصری را که مقدار آن برابر با ۳۹ هست را با استفاده از جستجوی خطی پیدا کنیم.

Searched Element  
39



13	9	21	15	39	19	27
0	1	2	3	4	5	6



## جستجوی دودویی

❖ الگوریتم جستجوی دودویی یک الگوریتم جستجوی است که در یک آرایه مرتب شده با تقسیم مکرر فاصله جستجوی به نصف استفاده می شود.

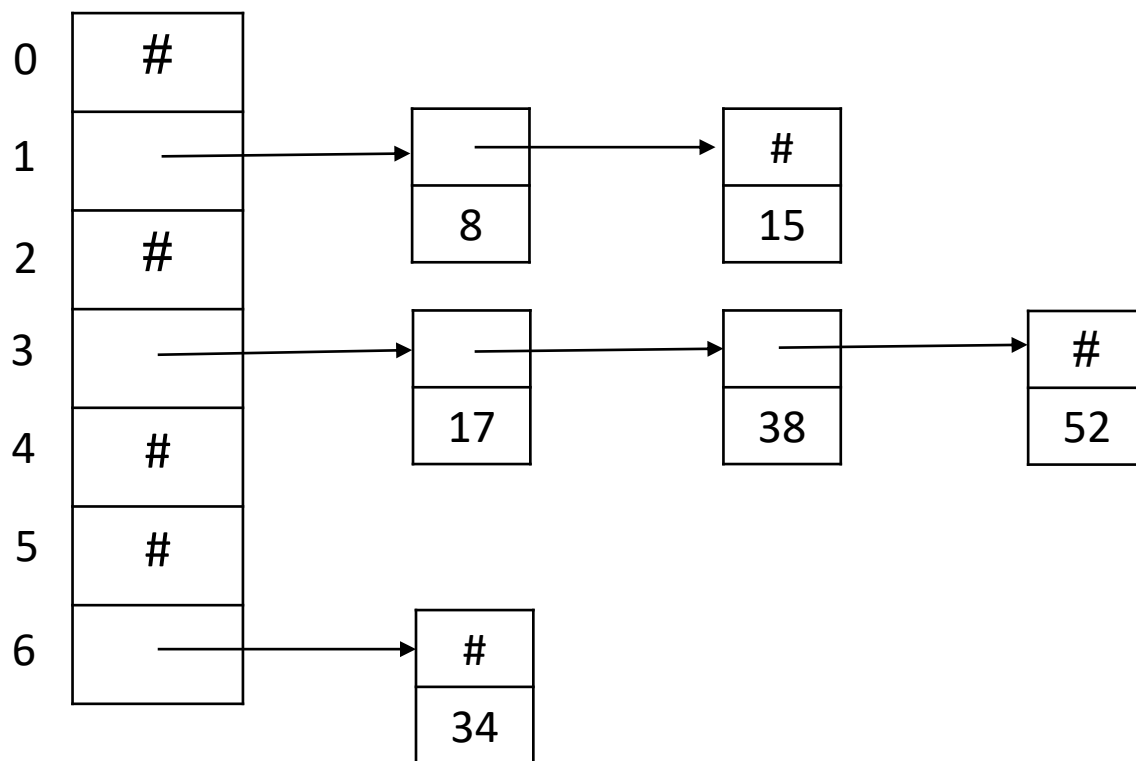
❖ آرایه زیر را در نظر بگیرید ، می خواهیم عنصری را که مقدار آن برابر با ۴۰ هست را با استفاده از جستجوی دودویی پیدا کنیم.

0	1	2	3	4	5	6	7	8	9
-10	-3	0	4	8	16	22	39	40	45
begin				mid		end			
-10	-3	0	4	8	16	22	39	40	45
					begin		mid		end
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	
								mid	
-10	-3	0	4	8	16	22	39	40	45
								begin	



## جدول درهم سازی

❖ جدول درهم سازی نوعی ساختار داده است که در آن مقدار آدرس عنصر از یک تابع هاش تولید می شود. این امکان دسترسی بسیار سریع به داده ها را فراهم می کند زیرا مقدار شاخص به عنوان یک کلید برای مقدار داده رفتار می کند



❖ در آرایه روبه رو هر خانه نشان دهنده یک لیست است. و هر عدد در لیستی قرار می گیرد که شماره آن لیست برابر باقی مانده آن عدد بر عدد ۷ است.



# 05.

## مسیر پوشای کمینه

Minimal path

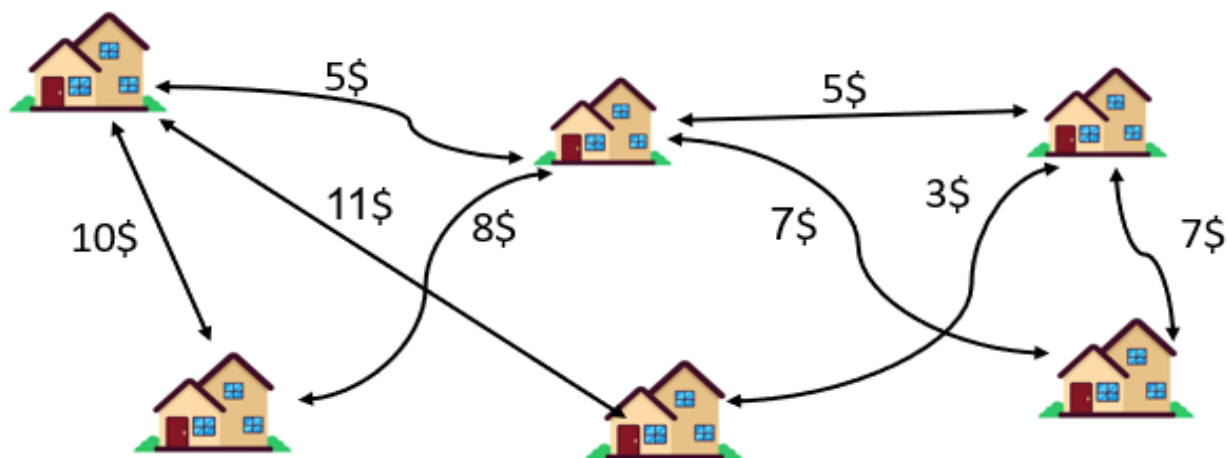




## کاربرد مسیر پوشای کمینه

❖ شبکه‌های زیادی اجتماعات ما را به هم پیوند می‌دهند: شبکه‌های تفلن، شبکه تامین آب، برق، گاز، شبکه‌های رایانه‌ای و شبکه‌های جاده‌ای، در هر شبکه، معمولا گزینه‌های مختلفی برای محل قرار گرفتن جاده‌ها، کابل‌ها یا ارتباطات رادیویی وجود دارد. ما باید راه‌های موثر را برای برقراری پیوند میان اجزای شبکه را پیدا کنیم.

❖ **مثال:** شهر زیر را در نظر بگیرید، همانطور که مشخص شده برای رفتن از خانه‌ای به خانه دیگر نیاز است که هزینه‌ای را پرداخت کنید، حال مسیری را پیدا کنید که با کمترین هزینه بتوان از یک خانه به تمام خانه‌ها رسید.



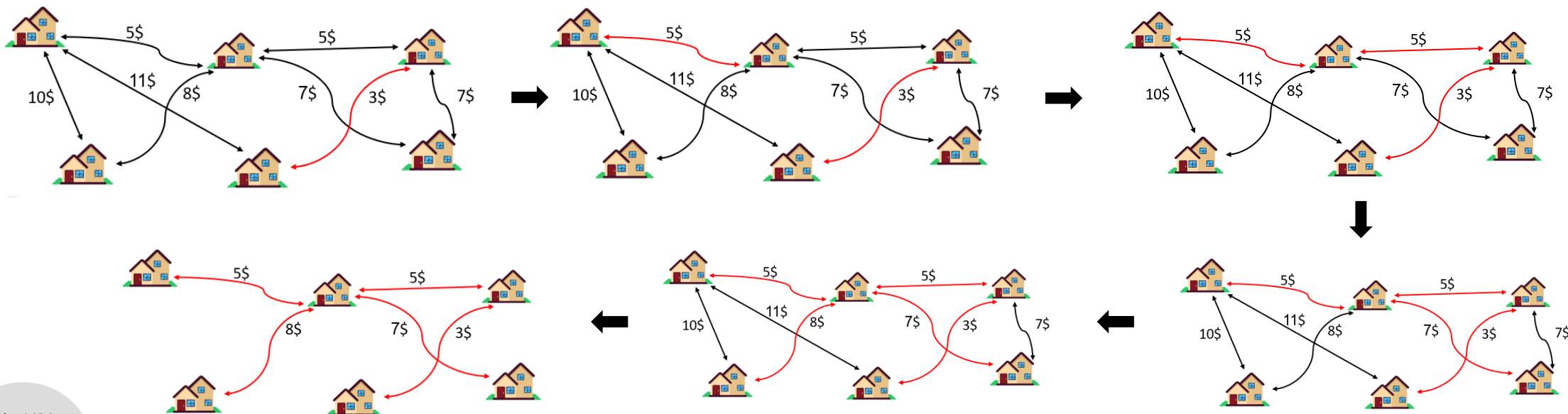
• آیا مسیری که پیدا کردید منحصر به فرد است؟



# الگوریتم پیدا کردن مسیر پوشای کمینه

❖ پیدا کردن درخت پوشای کمینه الگوریتم های زیادی دارد ، اما مرتبه زمانی هر کدام متفاوت است . الگوریتمی که قصد داریم ارائه دهیم، سریع ترین الگوریتم است.

❖ **الگوریتم:** در هر مرحله جاده‌ای را که کمترین هزینه را دارد انتخاب کنید و اگر قبلاً مسیری ، برای خانه های دو سر جاده وجود داشت آن جاده در نظر نمی گیریم در غیر این صورت آن جاده را انتخاب می‌کنیم





Talk is cheap, Show me the code.

Linus Torvalds.  
Programmer

[www.partschool.ir](http://www.partschool.ir)  
[www.part.institute](http://www.part.institute)  
[www.partsoftware.com](http://www.partsoftware.com)