

# پروژه دوم درس مقدمه ای بر هوش محاسباتی

## «حل مساله ی فروشنده ی دوره گرد به کمک الگوریتم ژنتیک»

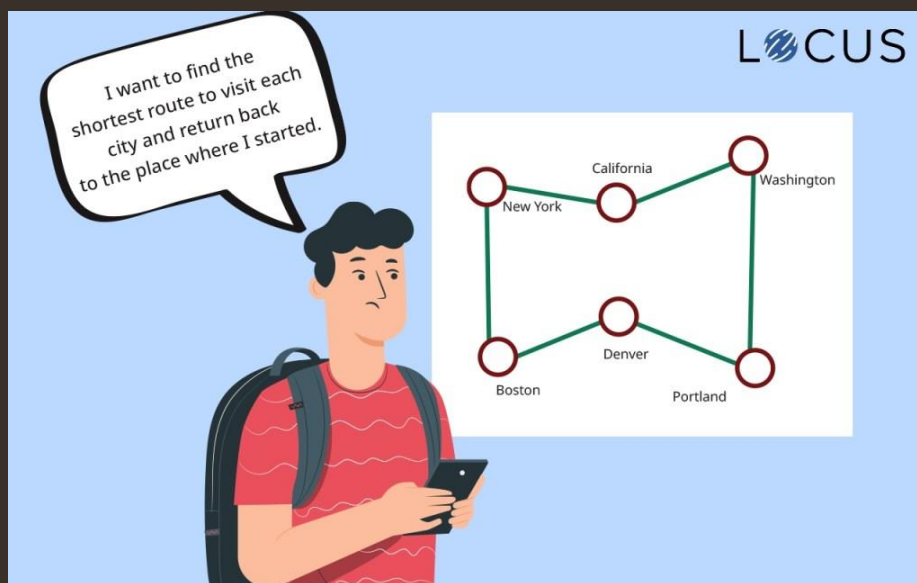
پاییز ۹۹

سارا خلیلی



# مسئله فروشنده دوره گرد

تعدادی شهر داریم و هزینه رفتن مستقیم از یکی به دیگری را می دانیم. مطلوب است کم هزینه ترین مسیری که از یک شهر شروع شود و از تمامی شهرها دقیقاً یکبار عبور کند و به شهر شروع بازگردد.



# کلیات الگوریتم ژنتیک

۱. انتخاب جمعیت اولیه
  ۲. محاسبه‌ی توابع برازندگی هر فرد در جمعیت کنونی
  ۳. انتخاب افراد برتر براساس برازندگی به عنوان والدین
  ۴. تولید نسل بعدی توسط عمل‌گرهای ژنتیکی
  ۵. محاسبه‌ی تابع برازندگی
- اگر شرایط خاتمه حاصل شده‌است پایان الگوریتم وگرنه برو به ۳.

# اصطلاحات الگوریتم

الف. **کد کردن پارامترها**، به صورت باینری ( $0$  و  $1$ ) و یا چندسطحی ( $M$ -ary) که بستگی به نوع مساله و تابع برازندگی دارد.

ب. **کروموزوم**، شامل پارامترهای (ژن‌های) مربوط به یک پاسخ ممکن برای مساله است. (طول آن وابسته به تعداد متغیرهای مساله و روش کد کردن است).

ج. **جمعیت ژنتیکی**، مجموعه‌ای از کروموزوم‌ها که در ابتدا انتخاب می‌شوند، تعداد اعضا ثابت است و یا می‌تواند متغیر باشد، مهم این است که همواره اعضا برتر در این جمعیت امکان بقا می‌یابند.

د. **تابع برازندگی**، بر روی جمعیت ژنتیکی اعمال می‌شود و از میان جمعیت آن‌هایی که برازندگی بالاتری داشته‌باشند امکان بقا می‌یابند و بقیه از دور خارج می‌شوند. یک راه عملی آن است که انتخاب افراد از جمعیت اولیه با احتمال متناسب با تابع برازندگی‌شان تقسیم بر مجموع برازندگی‌های کل جمعیت باشد، در نتیجه آن‌هایی که برازنده‌ترند بیشتر احتمال انتخاب دارند.

# عملگرهای ژنتیکی، برای تولید نسل بعدی از نسل قبل

- تولید مجدد (*Reproduction*) انتخاب کروموزومی مستقیماً بدون تغییر، از نسل فعلی به نسل بعدی براساس تابع برازندگی [پیشنهاد: ۴۹٪ از افراد جمعیت]
- جهش (*Mutation*) تغییر تصادفی یکی از بیت‌های والد (از ۰ به ۱ و بالعکس) [پیشنهاد: ۱٪]
- تقاطع (*Crossover*) تقاطع دو کروموزوم برای تولید کروموزومی احتمالاً بهتر به‌یکی از سه روش زیر:
  - تقاطع نقطه‌ای: شکستن دو کروموزوم از یک نقطه به‌طور تصادفی و تعویض قسمت‌های نظیر.
  - تقاطع دونقطه‌ای: شکستن کروموزوم‌ها از دو نقطه و تعویض قسمت‌های میانی نظیر.
  - تقاطع یکنواخت: استفاده از یک رشته‌ی تصادفی *MASK* برای انتخاب بیت‌های فرزند اولی یا دومی.

# پروژه دوم

۱- ۱۹ نقطه تصادفی در یک فضای سه بعدی در فاصله (۰ و ۰ و ۰) و (۱ و ۱ و ۱) به عنوان شهرها در مسئله فروشنده دوره گرد اختیار نمائید .

۲- مبدا مختصات را به عنوان شهر مبدأ انتخاب کرده و از آن می‌خواهیم به سایر شهرها هر کدام یکبار و فقط یکبار رفته و مجدداً "به مبدا" برگردیم . کوتاهترین مسیر مطلوبست .

۳- برای پیدا کردن ترتیب بهینه شهرها که مسیر طی شده در کل کمینه باشد ، از الگوریتم ژنتیک استفاده می‌کنیم . هر شهر را با یکی از حروف لاتین کد می‌کنیم .

۴- بهترین مسیر بدست آمده را در الگوریتم پس از تکرار اول ، تکرار صدم ، تکرار پانصدم و نیز در پایان الگوریتم روی صفحه رسم ( plot ) و مقدار بهینه را در هر مرحله تعیین و روند تکامل و بهینه‌سازی را مشاهده نمائید .

۵- مراحل کار خود و نتایج بدست آمده را توضیح دهید.

# قدم اول

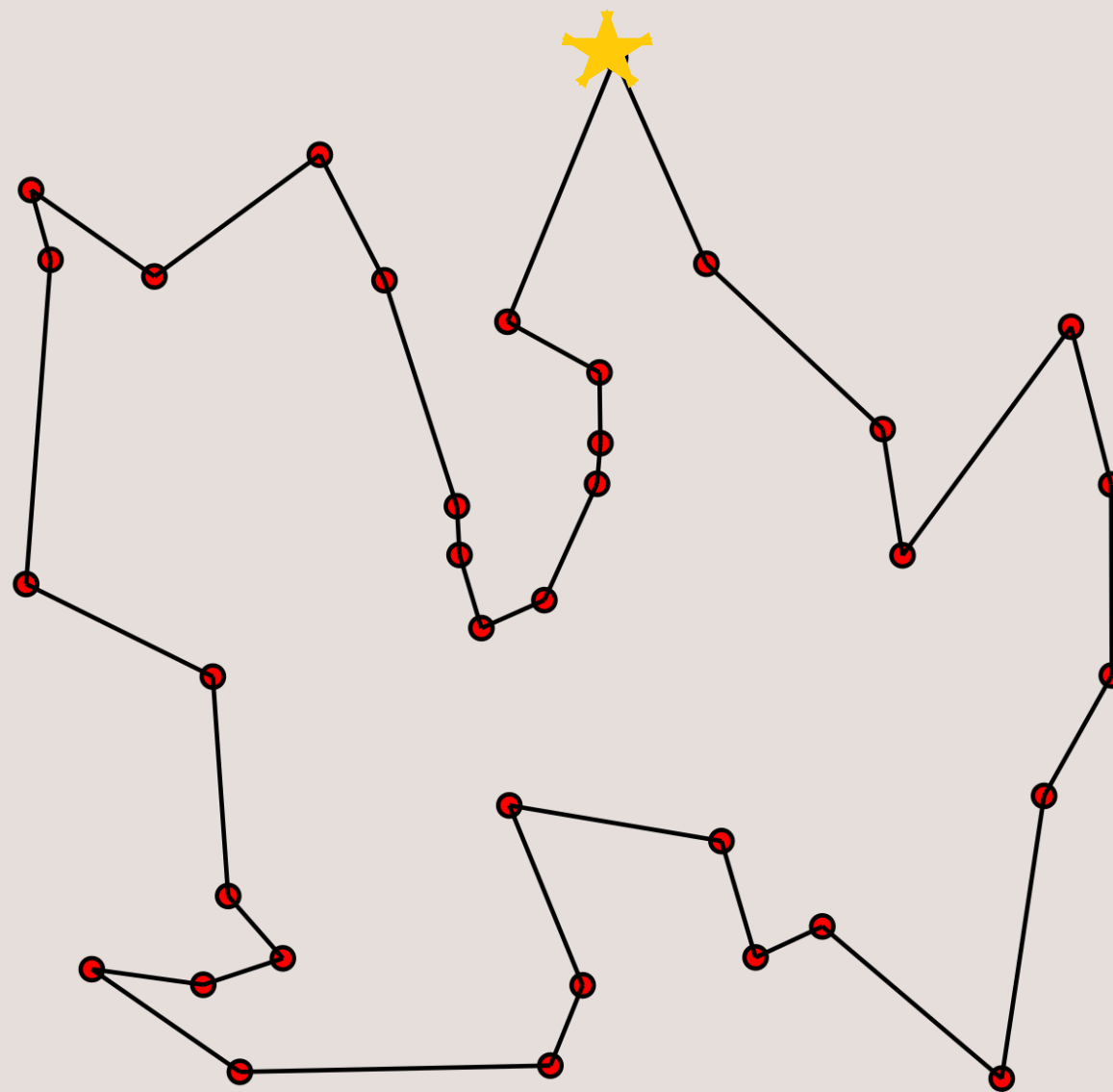
تولید شهرها

✓ ۲۰ شهر در حالت کلی

✓ ۱۹ شهر در کروموزوم

✓ ۲۰ فاصله بین شهری برای تابع هزینه

✓ ایجاد شهرها در مختصات دو بعدی به صورت رندوم



# قدم دوم

کد کردن شهر ها

✓ با استفاده از حروف لاتین

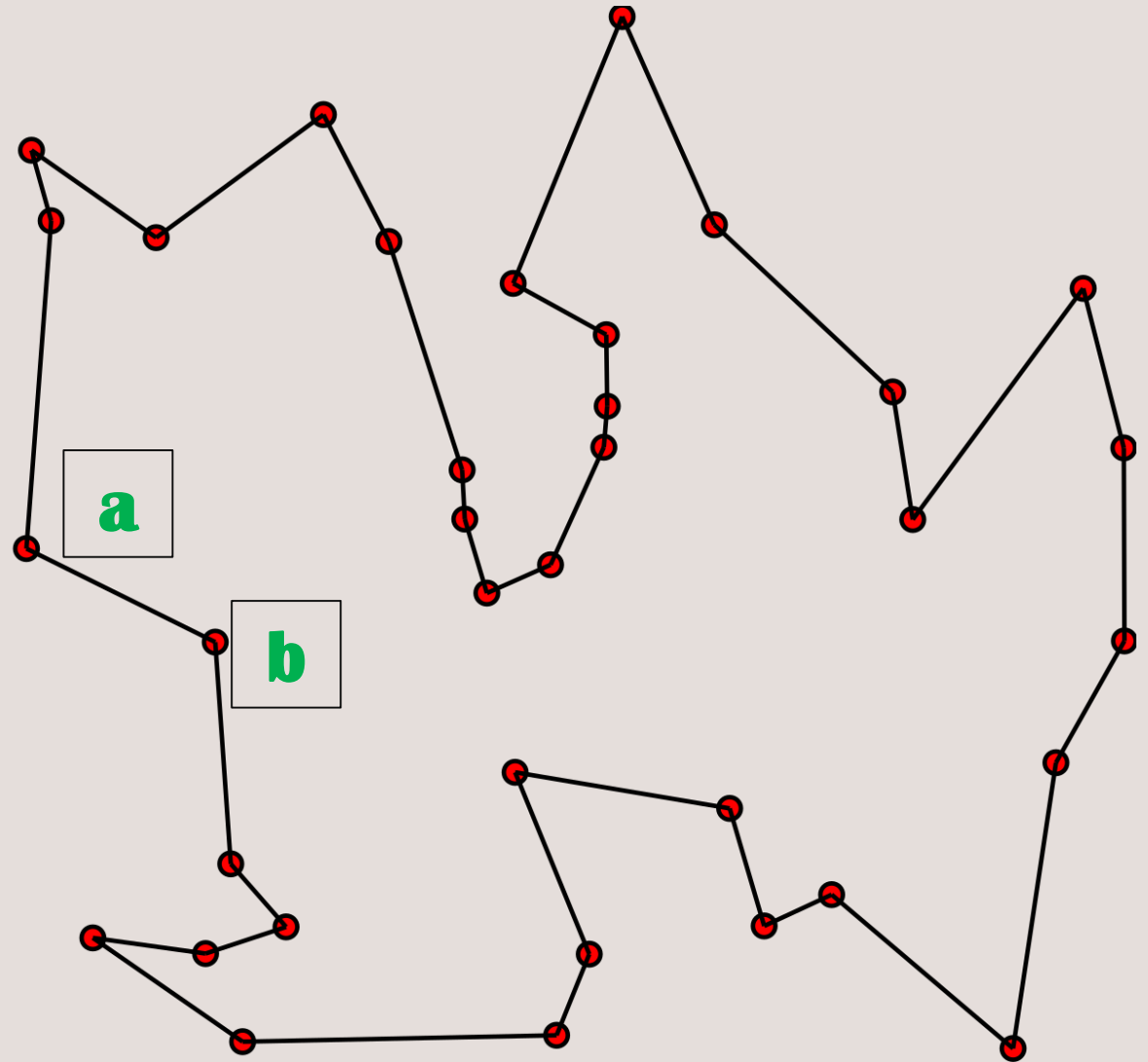
• برای مثال

$(x_1, y_1) \rightarrow a$

$(x_2, y_2) \rightarrow b$

✓ ایجاد ماتریس شهر ها به صورت معنا دار

$$x = \begin{bmatrix} 1 & x_1 & y_1 \\ 2 & x_2 & y_2 \\ 3 & x_3 & y_3 \\ \vdots & \vdots & \vdots \\ 19 & \vdots & \vdots \end{bmatrix}$$





# قدم سوم

ایجاد جمعیت ژنتیکی

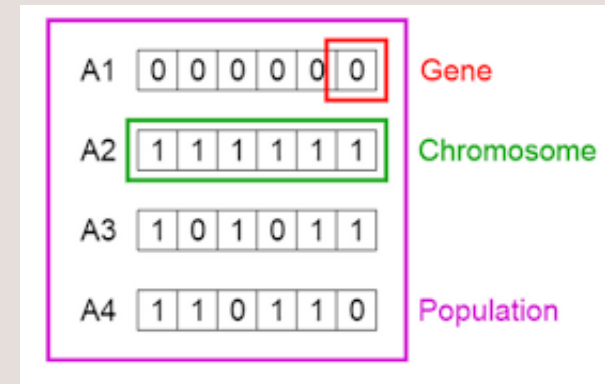
✓ عدم وجود ژن تکراری در کروموزوم

✓ تعداد کروموزوم ها  $\leq$  دلخواه

✓ Ch ماتریس کروموزوم ها

✓ n سطر نماینده n کروموزوم

✓ ۱۹ ستون نماینده ۱۹ شهر



Ch(1,:)	5	1	7	...	2	9	6	12	4
Ch(2,:)	1	6	3	...					
Ch(3,:)	:								
.									
.									
.									
Ch(i,:)									

$$ch(1,:) = random, i \in [1,19]$$



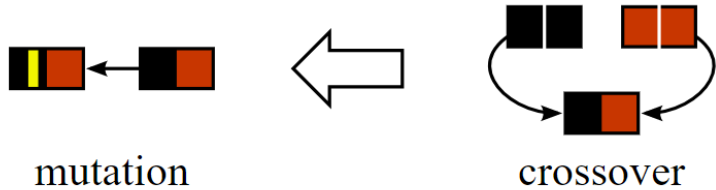
# قدم پنجم

## ایجاد نسل جدید:

توجه : برای اینکه عملگرهای ژنتیکی تقاطع و جهش در اینجا به ترتیب غیرمعتبر شهرها ( مسیر غیرمعتبر ) منجر نشود ، می توانید از عملگرهای زیر استفاده کنید :

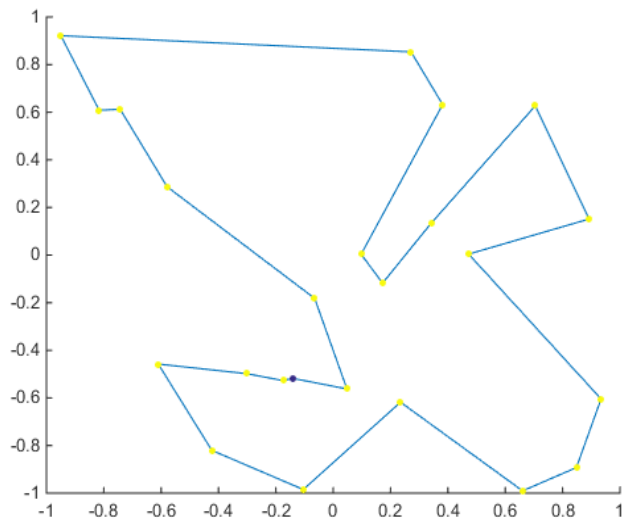
جهش : جابه جایی تصادفی دو شهر در ترتیب فعلی

تقاطع : معکوس کردن بخشی از زنجیره پس از بریدن تصادفی در نقطه ای از زنجیره .

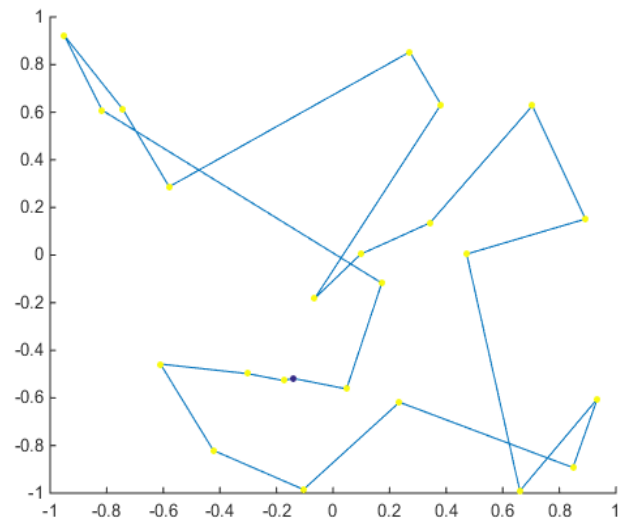


3	5	1	7	...	2	9	6	4
---	---	---	---	-----	---	---	---	---

## تکرار آخر



## تکرار پانصدم



## تکرار صدم

