تمرين دوم هوش محاسباتي به نام او عرفان عابدي، 9631427

یک.

 ۱. تنوع: به معنی ایجاد تفاوت در ویژگیهای افراد یک جمعیت است. اگر تنوع نداشتهباشیم همهی انتخابها میان نسلهایی با خصوصیتهای یکسان است و پیشرفتی حاصل نمی شود.

۲. انتخاب طبیعی: به معنی انتخاب بهترین و پرکاربردترین ژنها است. اگر انتخاب نداشتهباشیم جستوجویی صورت نمی گیرد
 تنوع به تنهایی موجب تکامل نمی شود و متوسط شایستگی را پایین می آورد، انتخاب را کنترل می کند و نمی گذارد که انتخاب تمام بهترینها را
 انتخاب کند. انتخاب نیز جلوی تنوع مداوم و فزاینده را می گیرد و صرفا انواع خاصی را به نسلهای بعد انتقال می دهد.

دو.

در حالت Generational کل جمعیت با نسل بعدی جایگزین می شود (از طریق انتخاب و تولیدمثل) ولی در Steady-state تنها تعداد محدودی از جمعیت در هر نسل جایگزین می شوند.

سه. در روش (u, l) برخلاف u + l ممکن است برخی ژنهای شایسته تر حذف شوند، اما خیلی دیر تر مجموعه ژنها را همگن می کند که برای گیر نکردن در local maxima ها بهتر است.

m =چهار. با فرض تعداد موجوداتی که می خواهیم انتخاب کنیم

$$\begin{split} \text{SP} &= \text{n * P}_{\text{best}} => \text{SP} = \text{m * (a + b * e^n)} = 1 => 1 = \text{m*a + m*e}^{n*b} \\ \text{We also know that the sum of all Pi's equals } 1 => \text{n.a + } b \sum_{s=1}^n e^i = \text{n.a + } \frac{b.e(e^n-1)}{e-1} = 1 \\ &=> \text{m*a + m*e}^{n*b} = \text{n.a + } \frac{b.e(e^n-1)}{e-1} = 1 \\ &=> \text{a = } b \frac{e^{n+1}-e-m.e^n}{(e-1)(m-n)} \text{, b = } a (\frac{e^{n+1}-e-m.e^n}{(e-1)(m-n)})^{\wedge} - 1 \end{split}$$

برای ۰ نشدن مخرج در صورتی که m=n باشد b باید ۰ باشد که در آن صورت داریم:

$$\mathsf{a} = \frac{e^{n+1} - e - n.e^n}{e - 1}$$

پنج. در صورت استفاده از u + l، وجود یک سیگمای نامناسب در یک فرد می تواند به علت شایستگی مناسب آن فرد برای مدت طولانی در جمعیت زنده بماند. در نتیجه روش u, l، کودتطبیقی بهتر است.

شش. چون X' و Sigma' جدید دوبار ارزیابی می شوند. ارزیابی اولیه: X' خوب است اگر X' خوب باشد. ارزیابی ثانویه: X' sigma' خوب است اگر X' خوب باشد.

هفت. لامارک اعتقاد داشت که تمامی صفات اکتسابی هر نسل به نسل بعد انتقال می یابد که منجر به تکامل می شود. داروین معتقد بود که تکامل از طریق انتخاب طبیعی و بقای adaptiveترین افراد شکل می گیرد.

علم مدرن به این نتیجه رسیده است که نظر داروین از صلاحیت علمی بیش تری برخوردار است، اما شایان ذکر است که بعضی خصوصیات اکتسابی (به ویژه در حوزه پاراژنتیک و محیط اطراف ژنهای انسان) امکان انتقال یک صفت اکتسابی از نسلی به نسل دیگر وجود دارد اما این تغییرات همیشگی نیستند و ممکن است با ایجاد تغییر در محیط ژنتیکی انسان، آن صفت اولیه دوباره در نسلها بروز کند.

هشت.

الف) احتمالی وجود دارد که هیچکدام از والدین شایسته برای تولیدمثل انتخاب نشوند. ب) تورنمنت بهتر است چون بایاس آن نسبت به بازماندگان شایسته کهتر اس. (جلوگیری از immature convergence) عرفان عابدي، 9631427 به نام او تمرین دوم هوش محاسباتی

پ) ممکن است عضوی که شایستگی ۲۵۰ را دارد همیشه انتخاب شود که به معنای ایجاد سکون در الگوریتم است چون همهی بازتر کیبی ها نتایج مشابه تولید خواهند کرد.

ت)

$$\begin{split} &f'_1 = a.f_i + b, \ f_b = 250, \ B = 3, \ \text{Avg}(f) = 29.1 \\ &=> f'_b = B.\text{avg}(f) = 58.2 \\ &=> a = \frac{f'_b - avg(f)}{f_b - avg(f)} = \frac{29.1}{250 - 29.1} = 0.132 \ , \ b = \frac{avg(f).(f_n - f'_b)}{f_b - avg(f)} = \frac{29.1 * (250 - 58.2)}{250 - 29.1} = 25.27 \\ &=> f'_i \ 0.132 * f_i + 25.27 \end{split}$$
 Therefore the new fitnesses will be:

26.3, 25.5, 26.1, 58.2, 25.8, 25.8, 26.2, 25.4, 25.8, 26.5

الف) تعداد ارزیابیهای انجام شده رابطه مستقیمی با تعداد فرزندان ایجاد شده در طی نسلها دارد و اندیکاتور بهتری برای میزان diverse بودن کروموزومهای تولید شده است.

ب) در هم *گ*رایی با ایجاد mutation نمی توانیم به مقادیر شایستگی جدیدی برسیم و همهی کروموزومها مقادیر یکسانی تولید می کنند اما در عدم تنوع تمامی افراد نسل مشابه میشوند و نمی توان با ایجاد تنوع کروموزوم جدیدی ایجاد کرد. ملاک اولی مقادیر شایستگی و ملاک دومی تعداد افراد جدید است.

مسالهي O-1 Knapsack:

- روش بازنمایی مساله: یک کروموزوم را یک لیست از اعداد ۰ یا ۱ تعریف می کنیم که ۱ بودن عضو ۱nم این لیست به معنای حضور جسم nام در انتخاب ما برای کوله پشتی است و بالعکس.
 - تعداد جمعیت = ۴۰
- تولید جمعیت اولیه و ارزیابی آنها: ابتدا همه ی کروموزومها را مجموعههای ۰ در نظر می گیریم، به این معنا که در هیچ کدام هیچ انتخابی نکرده ایم. پس مقدار fitness اولیهی همهی آنها ۰ است..
 - انتخاب والدین: با استفاده از روش roulette wheel به افرادی که fitness بیشتری دارند شانس بیشتری می دهیم.
- توليد فرزندان: با استفاده از one-point crossover ژنهاي والدين را تركيب مي كنيم و با استفاده از bit-flip mutation در ژنهای فرزندان جهش ایجاد می کنیم.
 - ارزیابی فرزندان: تابع fitness را معادل مجموع ارزشهای اشیا انتخاب شده می گذاریم به شرط این که مجموع وزنهای آنها از ظرفیت کولهپشتی بیش تر نشده باشد که در این صورت fitness صفر در نظر گرفته میشود.
 - انتخاب بازماندگان: با استفاده از fitness-based selection جوابهایی که در هر مرحله بهتر هستند را نگه می داریم.
- بررسی شرط خاتمه: یک عدد بزرگ را به عنوان ماکسیمم تعداد نسلها در نظر می گیریم و بعد از گذشت آن مقدار نسل، تکامل را متوقف مي كنيم.

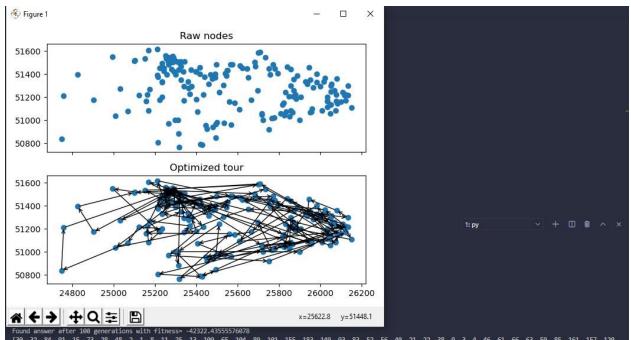
نتایج مساله 0-1 Knapsack برای تعداد نسلهای مختلف:

Found answer after 100 generations for file #1 with fitness= 8095		
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,	, 0, 0, 0, 0, 1, 0, 0, 0, 0,	0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,		
Found answer after 100 generations for file #2 with fitness= 1373		
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0	1, 0, 0, 0, 0, 0, 0, 0, 0, 0,	0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,		
Found answer after 100 generations for file #3 with fitness= 2065		
[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,	, 0, 0, 0, 0, 0, 0, 0, 0, 0,	0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0]		
Found answer after 200 generations for file #1 with fitness= 9147	. 10 (01) (02) (02) (1) (03)	MATRICE TO A MATRICE FOR 18 MATRICES OF MATRICES
[0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,	, 0, 0, 0, 0, 1, 0, 0, 0, 0,	0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,	, 0, 0, 0, 0, 1, 0, 0, 0, 0,	0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
[0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,	, 0, 0, 0, 0, 1, 0, 0, 0, 0,	0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,		
[0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,		
[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,		
[0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,	, 0, 0, 0, 0, 1, 0, 0, 1, 0,	8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 1, 8
[0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,	, 0, 0, 0, 0, 1, 0, 0, 1, 0,	8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 1, 8

مسالهى :TSP

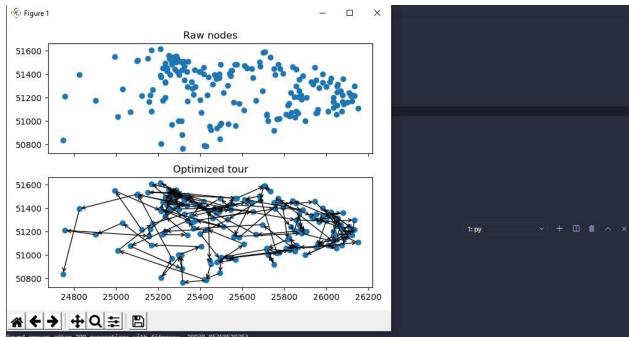
- روش بازنمایی مساله: یک کروموزوم را یک جایگشت از اعداد شهرها تعریف می کنیم که ترتیب این جایگشت ترتیب مسیرهای طی
 شده توسط فروشنده را نشان می دهد.
 - تعداد جمعیت = ۴۰
- تولید جمعیت اولیه و ارزیابی: ابتدا تعدادی جایگشت رندوم از شهرها تولید می کنیم و تابع fitness را برابر با منهای طول مسیر در نظر می گیریم.
- انتخاب والدین: با استفاده از روش Q-Tournament تعدادی را به صورت رندوم انتخاب می کنیم و هر بار از بین آنها بهترین فرد را برای تولیدمثل انتخاب می کنیم. لازم به ذکر است که قبل از اجرای تورنمنت الگوریتم Sigma Scaling را نیز اجرا می کنیم تا از هم گرایی زودرس جلوگیری شود.
 - تولید فرزندان: چون با یک کروموزوم جایگشتی مواجه هستیم از روش OX1 استفاده می کنیم برای ترکیب ژنهای والدین و از روش inverse mutation برای جهش دادن ژنهای فرزند استفاده می کنیم.
 - انتخاب بازماندگان: با استفاده از fitness-based selection جوابهایی که در هر مرحله بهتر هستند را نگه می داریم.
- بررسی شرط خاتمه: یک عدد بزرگ را به عنوان ماکسیمم تعداد نسلها در نظر می گیریم و بعد از گذشت آن مقدار نسل، تکامل را متوقف میکنیم.

نتايج الگوريتم مساله TSP براي تعداد نسلهاي مختلف:

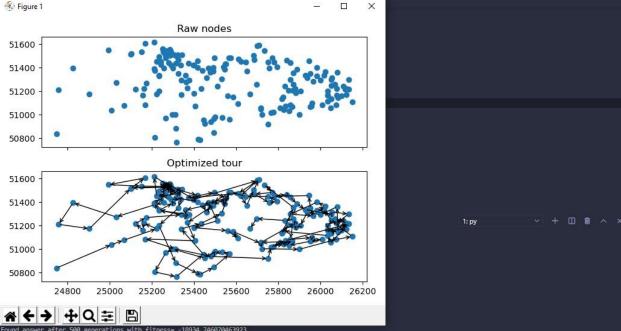


Found answer after 100 generations with fitness= -42322.43555576078

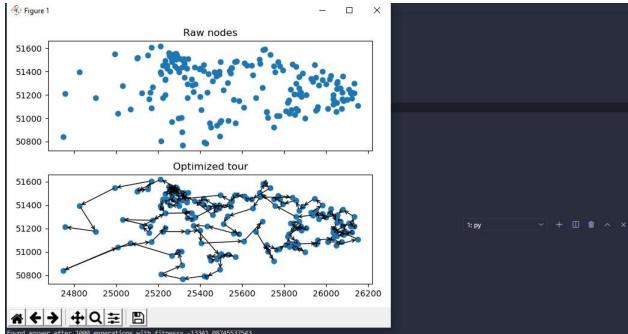
[30, 32, 84, 91, 15, 73, 28, 48, 2, 1, 8, 11, 25, 13, 109, 65, 104, 89, 101, 155, 183, 149, 93, 83, 52, 56, 40, 21, 22, 38, 9, 3, 4, 46, 61, 66, 63, 59, 85, 161, 157, 120, 53, 58, 31, 79, 75, 96, 97, 41, 35, 124, 148, 181, 190, 169, 116, 129, 177, 128, 92, 176, 187, 67, 50, 44, 12, 60, 55, 47, 7, 45, 24, 81, 77, 18, 88, 163, 139, 122, 132, 146, 126, 42, 123, 68, 72, 108, 105, 107, 106, 29, 49, 74, 69, 14, 16, 23, 5, 10, 39, 95, 64, 112, 162, 179, 165, 168, 170, 159, 160, 185, 193, 180, 136, 171, 191, 175, 19 2, 184, 131, 141, 164, 140, 144, 154, 150, 194, 189, 156, 127, 62, 102, 99, 125, 142, 172, 130, 36, 87, 51, 37, 34, 43, 54, 26, 78, 80, 6, 57, 110, 19, 27, 17, 76, 71, 103, 118, 151, 117, 121, 135, 174, 134, 111, 82, 113, 119, 143, 167, 178, 98, 86, 20, 137, 145, 147, 152, 138, 94, 90, 166, 186, 182, 153, 158, 188, 173, 114, 133, 115, 100, 70, 33]



Found answer after 200 generations with fitness= -29938.85358529753
[1, 3, 16, 36, 26, 65, 86, 90, 89, 88, 19, 56, 30, 37, 70, 9, 64, 14, 13, 2, 4, 7, 16, 82, 59, 63, 8, 72, 76, 122, 142, 149, 182, 189, 169, 176, 164, 179, 181, 178, 187, 156, 186, 171, 191, 188, 184, 190, 193, 165, 167, 185, 162, 158, 135, 113, 73, 165, 108, 69, 17, 68, 59, 80, 101, 125, 145, 139, 146, 137, 161, 173, 174, 175, 180, 159, 177, 166, 192, 183, 194, 163, 168, 147, 68, 55, 54, 57, 77, 38, 58, 52, 46, 44, 51, 39, 61, 41, 49, 48, 110, 93, 107, 155, 115, 120, 132, 134, 99, 80, 103, 121, 123, 128, 124, 112, 47, 42, 32, 35, 40, 53, 12, 43, 67, 81, 79, 119, 150, 157, 152, 153, 151, 144, 141, 131, 91, 83, 100, 106, 96, 92, 118, 78, 102, 18, 75, 28, 45, 15, 50, 31, 26, 71, 23, 25, 87, 114, 94, 111, 104, 140, 127, 130, 126, 138, 136, 143, 170, 160, 148, 129, 133, 172, 154, 109, 117, 116, 97, 60, 33, 24, 34, 27, 29, 66, 84, 74, 95, 22, 21, 5, 11]



Found answer after 500 generations with fitness= -18934.746070463923
[119, 122, 141, 152, 153, 126, 149, 140, 134, 137, 145, 161, 169, 163, 164, 146, 150, 154, 157, 159, 168, 185, 186, 173, 175, 181, 191, 183, 179, 174, 190, 187, 189, 188, 184, 193, 180, 178, 158, 155, 148, 143, 135, 160, 162, 165, 147, 166, 171, 167, 170, 177, 194, 182, 156, 127, 125, 113, 114, 109, 80, 102, 92, 93, 91, 88, 107, 105, 106, 1
10, 103, 87, 71, 89, 65, 20, 36, 59, 90, 101, 99, 104, 111, 98, 86, 85, 63, 25, 69, 70, 57, 60, 26, 78, 75, 74, 95, 97, 112, 117, 129, 136, 131, 121, 133, 128, 151, 138, 1
44, 139, 172, 176, 192, 142, 132, 130, 94, 62, 16, 82, 76, 96, 108, 100, 77, 115, 124, 120, 118, 116, 123, 79, 84, 67, 66, 34, 31, 35, 55, 41, 27, 37, 29, 45, 61, 73, 64,
83, 81, 49, 32, 42, 44, 43, 47, 53, 19, 22, 68, 39, 46, 10, 56, 58, 54, 48, 40, 30, 50, 51, 52, 38, 12, 5, 15, 9, 4, 2, 3, 7, 21, 33, 28, 18, 24, 72, 17, 11, 13, 14, 23, 8, 6, 1]



Found answer after 1000 generations with fitness= -13341.08745537543
[2, 4, 3, 5, 15, 12, 9, 10, 19, 50, 55, 54, 49, 42, 44, 35, 30, 32, 31, 27, 39, 34, 37, 40, 38, 48, 56, 53, 46, 41, 52, 43, 58, 47, 51, 64, 73, 67, 66, 61, 70, 68, 77, 91, 93, 96, 105, 106, 92, 88, 83, 81, 79, 84, 95, 97, 78, 75, 72, 69, 74, 26, 11, 13, 14, 7, 17, 24, 18, 21, 22, 28, 33, 60, 57, 29, 45, 100, 107, 108, 110, 112, 115, 116, 11
7, 128, 123, 124, 120, 131, 147, 159, 167, 168, 162, 165, 180, 185, 193, 189, 191, 192, 188, 174, 177, 171, 170, 178, 181, 184, 187, 190, 179, 186, 183, 173, 175, 194, 176
, 182, 169, 172, 164, 161, 163, 149, 146, 139, 152, 141, 154, 157, 144, 150, 153, 138, 140, 145, 156, 142, 137, 134, 132, 126, 125, 130, 127, 119, 122, 114, 82, 80, 71, 23
, 25, 16, 6, 1, 8, 36, 62, 59, 63, 20, 65, 85, 86, 98, 101, 111, 104, 94, 90, 89, 99, 76, 87, 109, 113, 102, 103, 118, 121, 143, 133, 129, 135, 136, 148, 151, 155, 160, 16
6, 158]

