گزارش پروژه ۲ درس هوش مصنوعی

آریا بنایی زاده ۹۴۳۱۰۲۹

هشت وزیر

موقعیت دهی y ملکه ها به صورت تصادفی داده می شود موقعیت x برای ملکه iام i است. عمل های موجود تغییر موقعیت y هر ملکه (یک خانه بالا یا یک خانه پایین) است.(الگوریتم ها هر بار یک خروجی را نمی دهند)

Normal Hill Climbing:

ابتدا سوال با normal hill climbing حل شد. در x , y موقعیت های ملکه ها در خروجی الگوریتم مشخص شده است. خروجی الگوریتم به شرح زیر است:

```
Normal Hill climbing...
expanding node  0  :
queen positions:  [[0, 7], [1, 6], [2, 0], [3, 4], [4, 3], [5, 7], [6, 7], [7, 0]]  Conflicting Values:  15
expanding node  7  :
queen positions:  [[0, 6], [1, 6], [2, 0], [3, 4], [4, 3], [5, 7], [6, 7], [7, 0]]  Conflicting Values:  10
expanding node  76  :
queen positions:  [[0, 6], [1, 6], [2, 0], [3, 4], [4, 1], [5, 7], [6, 7], [7, 0]]  Conflicting Values:  7
expanding node  131  :
queen positions:  [[0, 6], [1, 6], [2, 0], [3, 4], [4, 1], [5, 7], [6, 5], [7, 0]]  Conflicting Values:  5
expanding node  144  :
queen positions:  [[0, 6], [1, 3], [2, 0], [3, 4], [4, 1], [5, 7], [6, 5], [7, 0]]  Conflicting Values:  3
expanding node  212  :
queen positions:  [[0, 6], [1, 3], [2, 0], [3, 4], [4, 1], [5, 7], [6, 2], [7, 0]]  Conflicting Values:  3
expanding node  254  :
queen positions:  [[0, 6], [1, 3], [2, 0], [3, 4], [4, 1], [5, 5], [6, 2], [7, 0]]  Conflicting Values:  2
expanding node  276  :
queen positions:  [[0, 6], [1, 3], [2, 7], [3, 4], [4, 1], [5, 5], [6, 2], [7, 0]]  Conflicting Values:  1
did not found the right Answer exiting Now ...
```

همانطور که در خروجی می بینیم الگوریتم کامل نیست و جواب درست را به ما نمی دهد.
(متغیر conflicting values تعداد بار هایی است که دو ملکه همدیگر را تحدید می کنند)

Stochastic Hill Climbing:

خروجی الگوریتم:

```
Stochastic Hill climbing...
expanding node  0  :
queen positions:  [[0, 2], [1, 7], [2, 2], [3, 2], [4, 6], [5, 0], [6, 1], [7, 7]]  Conflicting Values:  8
expanding node  1  :
```

queen positions: [[0, 3], [1, 7], [2, 2], [3, 2], [4, 6], [5, 0], [6, 1], [7, 7]] Conflicting Values: 5
expanding node 69 :
queen positions: [[0, 3], [1, 7], [2, 2], [3, 0], [4, 6], [5, 0], [6, 1], [7, 7]] Conflicting Values: 5
expanding node 106 :
queen positions: [[0, 3], [1, 7], [2, 4], [3, 0], [4, 6], [5, 0], [6, 1], [7, 7]] Conflicting Values: 5
expanding node 162 :
queen positions: [[0, 3], [1, 7], [2, 4], [3, 0], [4, 6], [5, 3], [6, 1], [7, 7]] Conflicting Values: 5
expanding node 186 :
queen positions: [[0, 3], [1, 5], [2, 4], [3, 0], [4, 6], [5, 3], [6, 1], [7, 7]] Conflicting Values: 4
expanding node 225 :
queen positions: [[0, 3], [1, 5], [2, 0], [3, 0], [4, 6], [5, 3], [6, 1], [7, 7]] Conflicting Values: 4
expanding node 285 :
queen positions: [[0, 3], [1, 5], [2, 0], [3, 0], [4, 6], [5, 6], [6, 1], [7, 7]] Conflicting Values: 3
expanding node 297 :
queen positions: [[0, 4], [1, 5], [2, 0], [3, 0], [4, 6], [5, 6], [6, 1], [7, 7]] Conflicting Values: 3
expanding node 353 :
queen positions: [[0, 4], [1, 5], [2, 0], [3, 2], [4, 6], [5, 6], [6, 1], [7, 7]] Conflicting Values: 2
expanding node 410 :
queen positions: [[0, 4], [1, 5], [2, 0], [3, 2], [4, 6], [5, 3], [6, 1], [7, 7]] Conflicting Values: 2
expanding node 430 :
queen positions: [[0, 4], [1, 2], [2, 0], [3, 2], [4, 6], [5, 3], [6, 1], [7, 7]] Conflicting Values: 2
expanding node 473 :
queen positions: [[0, 4], [1, 2], [2, 7], [3, 2], [4, 6], [5, 3], [6, 1], [7, 7]] Conflicting Values: 2
expanding node 541 :
queen positions: [[0, 4], [1, 2], [2, 7], [3, 2], [4, 6], [5, 3], [6, 0], [7, 7]] Conflicting Values: 2
expanding node 555 :
queen positions: [[0, 4], [1, 1], [2, 7], [3, 2], [4, 6], [5, 3], [6, 0], [7, 7]] Conflicting Values: 2
expanding node 595 :
queen positions: [[0, 4], [1, 1], [2, 5], [3, 2], [4, 6], [5, 3], [6, 0], [7, 7]] Conflicting Values: 1
did not found the right Answer exiting Now ...

این الگوریتم نیز مانند الگوریتم بالا کامل نیست
در این روش ابتدا گره های بچه را ایجاد میکنیم و سپس به صورت تصادفی از گره هایی که باعث صعود می شوند یکی را انتخاب می کنیم.

First Choice Hill Climbing:

خروجی الگوریتم:

first choice Hill climbing...
expanding node 0 :
queen positions: [[0, 0], [1, 1], [2, 4], [3, 7], [4, 0], [5, 2], [6, 7], [7, 4]] Conflicting Values: 5
expanding node 1 :
queen positions: [[0, 1], [1, 1], [2, 4], [3, 7], [4, 0], [5, 2], [6, 7], [7, 4]] Conflicting Values: 5
expanding node 85 :

queen positions: [[0, 1], [1, 1], [2, 4], [3, 7], [4, 0], [5, 2], [6, 0], [7, 4]] Conflicting Values: 5
expanding node 93 :
queen positions: [[0, 3], [1, 1], [2, 4], [3, 7], [4, 0], [5, 2], [6, 0], [7, 4]] Conflicting Values: 4
expanding node 158 :
queen positions: [[0, 3], [1, 1], [2, 4], [3, 7], [4, 5], [5, 2], [6, 0], [7, 4]] Conflicting Values: 3
expanding node 189 :
queen positions: [[0, 3], [1, 1], [2, 7], [3, 7], [4, 5], [5, 2], [6, 0], [7, 4]] Conflicting Values: 3
expanding node 231 :
queen positions: [[0, 3], [1, 1], [2, 7], [3, 2], [4, 5], [5, 2], [6, 0], [7, 4]] Conflicting Values: 3
expanding node 283 :
queen positions: [[0, 3], [1, 1], [2, 7], [3, 2], [4, 5], [5, 3], [6, 0], [7, 4]] Conflicting Values: 2
expanding node 326 :
queen positions: [[0, 3], [1, 1], [2, 7], [3, 2], [4, 0], [5, 3], [6, 0], [7, 4]] Conflicting Values: 2
expanding node 340 :
queen positions: [[0, 6], [1, 1], [2, 7], [3, 2], [4, 0], [5, 3], [6, 0], [7, 4]] Conflicting Values: 2
expanding node 391 :
queen positions: [[0, 6], [1, 1], [2, 5], [3, 2], [4, 0], [5, 3], [6, 0], [7, 4]] Conflicting Values: 2
expanding node 422 :
queen positions: [[0, 1], [1, 1], [2, 5], [3, 2], [4, 0], [5, 3], [6, 0], [7, 4]] Conflicting Values: 2
expanding node 501 :
queen positions: [[0, 1], [1, 1], [2, 5], [3, 2], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 2
expanding node 503 :
queen positions: [[0, 3], [1, 1], [2, 5], [3, 2], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 2
expanding node 555 :
queen positions: [[0, 3], [1, 1], [2, 4], [3, 2], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 2
expanding node 584 :
queen positions: [[0, 5], [1, 1], [2, 4], [3, 2], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 2
expanding node 639 :
queen positions: [[0, 5], [1, 1], [2, 4], [3, 6], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
expanding node 681 :
queen positions: [[0, 5], [1, 1], [2, 1], [3, 6], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
expanding node 715 :
queen positions: [[0, 5], [1, 5], [2, 1], [3, 6], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
expanding node 748 :
queen positions: [[0, 0], [1, 5], [2, 1], [3, 6], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
expanding node 812 :
queen positions: [[0, 0], [1, 5], [2, 1], [3, 6], [4, 6], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
expanding node 851 :
queen positions: [[0, 0], [1, 5], [2, 1], [3, 2], [4, 6], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
expanding node 888 :
queen positions: [[0, 0], [1, 5], [2, 7], [3, 2], [4, 6], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
final positions in node 945 : [[0, 0], [1, 5], [2, 7], [3, 2], [4, 6], [5, 3], [6, 1], [7, 4]] Conflicting Values: 0

این الگوریتم نیز کامل نیست اما در خروجی بالا به جواب رسیده است.
در این روش ابتدا تا زمانی که گره بچه ایجاد شده باعث ایجاد حالت بهتر نشود گره جدید ایجاد می کنیم در غیر این صورت گره موجود را برابر با گره بچه می گذاریم.
مشکل این سه الگوریتم این است که اگر به مقداری غیر از جواب میل کنند در آن گیر می کنند.(ماکسیمای محلی)

Random Restart Hill Climbing:

خروجی الگوریتم در فایل ضمیمه آمده است.
این الگوریتم کامل است بعد از هر همگرایی از جایی دیگر به صورت تصادفی آغاز می کنیم. اگر تعریف کنیم بعد از n گسترش الگوریتم تمام شود ممکن است به جواب نرسد. حالت جواب زیر با n=100 داده شده که به جواب رسیده است.

first choice Hill climbing...
expanding node 0 :
queen positions: [[0, 0], [1, 1], [2, 4], [3, 7], [4, 0], [5, 2], [6, 7], [7, 4]] Conflicting Values: 5
expanding node 1 :
queen positions: [[0, 1], [1, 1], [2, 4], [3, 7], [4, 0], [5, 2], [6, 7], [7, 4]] Conflicting Values: 5
expanding node 85 :
queen positions: [[0, 1], [1, 1], [2, 4], [3, 7], [4, 0], [5, 2], [6, 0], [7, 4]] Conflicting Values: 5
expanding node 93 :
queen positions: [[0, 3], [1, 1], [2, 4], [3, 7], [4, 0], [5, 2], [6, 0], [7, 4]] Conflicting Values: 4
expanding node 158 :
queen positions: [[0, 3], [1, 1], [2, 4], [3, 7], [4, 5], [5, 2], [6, 0], [7, 4]] Conflicting Values: 3
expanding node 189 :
queen positions: [[0, 3], [1, 1], [2, 7], [3, 7], [4, 5], [5, 2], [6, 0], [7, 4]] Conflicting Values: 3
expanding node 231 :
queen positions: [[0, 3], [1, 1], [2, 7], [3, 2], [4, 5], [5, 2], [6, 0], [7, 4]] Conflicting Values: 3
expanding node 283 :
queen positions: [[0, 3], [1, 1], [2, 7], [3, 2], [4, 5], [5, 3], [6, 0], [7, 4]] Conflicting Values: 2
expanding node 326 :
queen positions: [[0, 3], [1, 1], [2, 7], [3, 2], [4, 0], [5, 3], [6, 0], [7, 4]] Conflicting Values: 2
expanding node 340 :
queen positions: [[0, 6], [1, 1], [2, 7], [3, 2], [4, 0], [5, 3], [6, 0], [7, 4]] Conflicting Values: 2
expanding node 391 :
queen positions: [[0, 6], [1, 1], [2, 5], [3, 2], [4, 0], [5, 3], [6, 0], [7, 4]] Conflicting Values: 2
expanding node 422 :
queen positions: [[0, 1], [1, 1], [2, 5], [3, 2], [4, 0], [5, 3], [6, 0], [7, 4]] Conflicting Values: 2
expanding node 501 :
queen positions: [[0, 1], [1, 1], [2, 5], [3, 2], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 2
expanding node 503 :
queen positions: [[0, 3], [1, 1], [2, 5], [3, 2], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 2
expanding node 555 :
queen positions: [[0, 3], [1, 1], [2, 4], [3, 2], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 2
expanding node 584 :
queen positions: [[0, 5], [1, 1], [2, 4], [3, 2], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 2

expanding node 639 :
queen positions: [[0, 5], [1, 1], [2, 4], [3, 6], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
expanding node 681 :
queen positions: [[0, 5], [1, 1], [2, 1], [3, 6], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
expanding node 715 :
queen positions: [[0, 5], [1, 5], [2, 1], [3, 6], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
expanding node 748 :
queen positions: [[0, 0], [1, 5], [2, 1], [3, 6], [4, 0], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
expanding node 812 :
queen positions: [[0, 0], [1, 5], [2, 1], [3, 6], [4, 6], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
expanding node 851 :
queen positions: [[0, 0], [1, 5], [2, 1], [3, 2], [4, 6], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
expanding node 888 :
queen positions: [[0, 0], [1, 5], [2, 7], [3, 2], [4, 6], [5, 3], [6, 7], [7, 4]] Conflicting Values: 1
final positions in node 945 : [[0, 0], [1, 5], [2, 7], [3, 2], [4, 6], [5, 3], [6, 1], [7, 4]] Conflicting Values: 0

قسمت بندی گراف:

برای حل قسمت بندی گراف مقدار مینیمم تابع زیر را با استفاده از الگوریتم simmulated annealing می یابیم:
اندازه تفاضل تعداد یال های هر بخش تقسیم بر دو + تعداد یال های وصل کننده دو قسمت به هم
برای اجرای simulated Annealing از سه روش کاهش خطی و نمایی و ضریبی استفاده شده است.
خطی: بعد از هر بار گسترش مقدار مشخصی از p کم میکنیم
خروجی الگوریتم(توجه شود پیدا کردن جواب مناسب بسته به حداقل مقدار مورد انتظار ما متفاوت است):

خطی:

simulated Annealing...
sub node: [] Cost: 8.5 p: 1.0
current Node: 3
sub node: [2] Cost: 10.0 p: 0.4999
current Node: 19
sub node: [2, 7] Cost: 12.0 p: 0.24985000000000002
current Node: 33
sub node: [2, 7, 11] Cost: 11.0 p: 0.124825
current Node: 35

```
sub node:  [2, 7, 11, 1]  Cost:  11.0  p: 0.0623125
current Node:  46
sub node:  [2, 7, 11, 1, 4]  Cost:  8.5  p: 0.03105625
current Node:  63
sub node:  [2, 7, 1, 4]  Cost:  9.5  p: 0.015428125
current Node:  64
sub node:  [2, 7, 1, 4, 0]  Cost:  6.5  p: 0.0076140625
current Node:  75
sub node:  [2, 7, 1, 4, 0, 5]  Cost:  6.0  p: 0.0037070312500000003
current Node:  94
sub node:  [2, 1, 4, 0, 5]  Cost:  4.0  p: 0.001753515625
current Node:  95
final Partitions:  [2, 1, 4, 0, 5, 3] [6, 7, 8, 9, 10, 11]
```

ضریبی:

```
simulated Annealing...
sub node:  []  Cost:  8.5  p: 1.0
current Node:  3
sub node:  [2]  Cost:  10.0  p: 0.4999
current Node:  19
sub node:  [2, 7]  Cost:  12.0  p: 0.24985000000000002
current Node:  33
sub node:  [2, 7, 11]  Cost:  11.0  p: 0.124825
current Node:  35
sub node:  [2, 7, 11, 1]  Cost:  11.0  p: 0.0623125
current Node:  46
sub node:  [2, 7, 11, 1, 4]  Cost:  8.5  p: 0.03105625
current Node:  63
sub node:  [2, 7, 1, 4]  Cost:  9.5  p: 0.015428125
current Node:  64
sub node:  [2, 7, 1, 4, 0]  Cost:  6.5  p: 0.0076140625
current Node:  75
sub node:  [2, 7, 1, 4, 0, 5]  Cost:  6.0  p: 0.0037070312500000003
current Node:  94
sub node:  [2, 1, 4, 0, 5]  Cost:  4.0  p: 0.001753515625
current Node:  95
final Partitions:  [2, 1, 4, 0, 5, 3] [6, 7, 8, 9, 10, 11] Cost: 3.0
```

نمایی:

```
simulated Annealing...
sub node:  []  Cost:  6.800000000000001  p: 1.0
```

current Node: 4
sub node: [3] Cost: 8.0 p: 0.9997999799986668
current Node: 16
sub node: [3, 4] Cost: 9.8 p: 0.9995998999799988
current Node: 24
sub node: [3, 4, 0] Cost: 11.0 p: 0.9993997198839691
current Node: 34
sub node: [3, 4, 0, 1] Cost: 7.4 p: 0.9991993995744312
current Node: 46
sub node: [3, 4, 0, 1, 6] Cost: 7.8 p: 0.9989988988069002
current Node: 58
sub node: [3, 4, 0, 1, 6, 9] Cost: 7.0 p: 0.9987981771961557
current Node: 69
sub node: [3, 4, 0, 1, 6, 9, 11] Cost: 9.8 p: 0.9985971941836088
current Node: 79
sub node: [3, 4, 1, 6, 9, 11] Cost: 9.0 p: 0.9983959090043787
current Node: 93
sub node: [3, 4, 1, 9, 11] Cost: 10.6 p: 0.9981942806540203
current Node: 99
sub node: [3, 4, 1, 9, 11, 7] Cost: 9.0 p: 0.9979922678548476
current Node: 116
sub node: [3, 4, 1, 11, 7] Cost: 7.8 p: 0.9977898290217893
current Node: 119
sub node: [3, 4, 1, 11, 7, 2] Cost: 8.4 p: 0.9975869222277194
current Node: 135
sub node: [3, 4, 11, 7, 2] Cost: 11.2 p: 0.9973835051682006
current Node: 149
sub node: [3, 4, 11, 2] Cost: 12.8 p: 0.9971795351255739
current Node: 151
sub node: [3, 4, 11, 2, 0] Cost: 12.0 p: 0.9969749689323316
current Node: 161
sub node: [3, 4, 11, 2, 0, 1] Cost: 6.4 p: 0.9967697629337033
current Node: 174
sub node: [3, 4, 11, 2, 0, 1, 7] Cost: 7.2 p: 0.9965638729493864
current Node: 186
sub node: [3, 4, 11, 2, 0, 1, 7, 10] Cost: 9.4 p: 0.9963572542343466
current Node: 194
sub node: [3, 4, 11, 2, 0, 1, 7, 10, 8] Cost: 10.2 p: 0.9961498614386135
current Node: 207
sub node: [3, 4, 11, 0, 1, 7, 10, 8] Cost: 10.0 p: 0.9959416485659915
current Node: 218
sub node: [4, 11, 0, 1, 7, 10, 8] Cost: 9.2 p: 0.9957325689316048
current Node: 225

sub node: [4, 11, 0, 1, 7, 10, 8, 5] Cost: 9.4 p: 0.995522575118189
current Node: 241
sub node: [11, 0, 1, 7, 10, 8, 5] Cost: 11.2 p: 0.9953116189310405
current Node: 253
sub node: [11, 0, 1, 10, 8, 5] Cost: 10.4 p: 0.9950996513515288
current Node: 261
sub node: [11, 0, 1, 10, 8, 5, 7] Cost: 11.2 p: 0.9948866224890727
current Node: 277
sub node: [11, 0, 1, 10, 5, 7] Cost: 12.4 p: 0.9946724815314762
current Node: 287
sub node: [11, 0, 10, 5, 7] Cost: 11.2 p: 0.9944571766935164
current Node: 296
sub node: [11, 0, 10, 5, 7, 8] Cost: 8.4 p: 0.9942406551636641
current Node: 308
sub node: [11, 0, 10, 5, 7, 8, 9] Cost: 8.4 p: 0.9940228630488213
current Node: 314
sub node: [11, 0, 10, 5, 7, 8, 9, 1] Cost: 12.0 p: 0.9938037453169412
current Node: 327
sub node: [11, 0, 10, 5, 7, 8, 9, 1, 6] Cost: 11.6 p: 0.9935832457373995
current Node: 338
sub node: [11, 0, 10, 7, 8, 9, 1, 6] Cost: 9.4 p: 0.993361306818972
current Node: 351
sub node: [11, 0, 10, 7, 9, 1, 6] Cost: 8.6 p: 0.9931378697452642
current Node: 355
sub node: [11, 0, 10, 7, 9, 1, 6, 2] Cost: 8.8 p: 0.9929128743074369
current Node: 372
sub node: [11, 0, 10, 7, 9, 6, 2] Cost: 7.2 p: 0.9926862588340505
current Node: 383
sub node: [11, 10, 7, 9, 6, 2] Cost: 6.4 p: 0.9924579601178518
current Node: 390
sub node: [11, 10, 7, 9, 6, 2, 1] Cost: 10.0 p: 0.9922279133393067
current Node: 410
sub node: [11, 7, 9, 6, 2, 1] Cost: 11.8 p: 0.9919960519866742
current Node: 417
sub node: [11, 7, 9, 6, 2, 1, 10] Cost: 10.0 p: 0.9917623077724046
current Node: 424
sub node: [11, 7, 9, 6, 2, 1, 10, 3] Cost: 8.8 p: 0.991526610545626
current Node: 440
sub node: [11, 7, 9, 2, 1, 10, 3] Cost: 11.2 p: 0.9912888882004733
current Node: 451
sub node: [11, 9, 2, 1, 10, 3] Cost: 8.4 p: 0.9910490665799925
current Node: 460
sub node: [11, 9, 2, 1, 10, 3, 8] Cost: 10.4 p: 0.9908070693753377

current Node: 467
sub node: [11, 9, 2, 1, 10, 3, 8, 0] Cost: 9.2 p: 0.9905628180199579
current Node: 480
sub node: [11, 9, 2, 1, 10, 3, 8, 0, 6] Cost: 8.8 p: 0.9903162315784503
current Node: 491
sub node: [11, 9, 2, 10, 3, 8, 0, 6] Cost: 9.2 p: 0.9900672266297317
current Node: 504
sub node: [11, 9, 2, 10, 8, 0, 6] Cost: 8.4 p: 0.9898157171441584
current Node: 518
sub node: [11, 2, 10, 8, 0, 6] Cost: 10.4 p: 0.9895616143541947
current Node: 527
sub node: [11, 2, 10, 8, 6] Cost: 11.2 p: 0.9893048266182048
current Node: 538
sub node: [11, 2, 10, 8, 6, 7] Cost: 8.4 p: 0.9890452592769071
current Node: 551
sub node: [11, 10, 8, 6, 7] Cost: 5.8 p: 0.9887828145019997
current Node: 564
sub node: [11, 10, 6, 7] Cost: 6.6 p: 0.9885173911364266
current Node: 569
sub node: [11, 10, 6, 7, 2] Cost: 8.4 p: 0.9882488845257154
current Node: 585
sub node: [11, 10, 6, 7, 2, 9] Cost: 6.4 p: 0.9879771863397708
current Node: 594
sub node: [11, 10, 6, 7, 2, 9, 5] Cost: 8.6 p: 0.9877021843844644
current Node: 600
sub node: [11, 10, 6, 7, 2, 9, 5, 0] Cost: 9.4 p: 0.9874237624023051
current Node: 617
sub node: [11, 10, 6, 7, 2, 5, 0] Cost: 10.6 p: 0.9871417998614227
current Node: 622
sub node: [11, 10, 6, 7, 2, 5, 0, 4] Cost: 10.8 p: 0.9868561717320309
current Node: 635
sub node: [11, 10, 6, 7, 5, 0, 4] Cost: 10.6 p: 0.9865667482494709
current Node: 644
sub node: [11, 10, 6, 7, 5, 0, 4, 8] Cost: 9.4 p: 0.9862733946628645
current Node: 656
sub node: [11, 10, 6, 7, 5, 4, 8] Cost: 6.6 p: 0.9859759709683212
current Node: 666
sub node: [11, 10, 6, 7, 5, 4, 8, 3] Cost: 7.4 p: 0.9856743316255601
current Node: 674
sub node: [11, 10, 6, 7, 5, 4, 8, 3, 0] Cost: 10.2 p: 0.9853683252567073
current Node: 692
sub node: [11, 10, 6, 7, 5, 4, 3, 0] Cost: 11.4 p: 0.9850577943259267
current Node: 703

sub node: [11, 10, 6, 5, 4, 3, 0] Cost: 11.8 p: 0.984742574798424
current Node: 717
sub node: [10, 6, 5, 4, 3, 0] Cost: 11.0 p: 0.9844224957772365
current Node: 727
sub node: [10, 6, 4, 3, 0] Cost: 11.2 p: 0.9840973791160815
current Node: 734
sub node: [10, 6, 4, 3, 0, 8] Cost: 12.4 p: 0.9837670390063824
current Node: 745
sub node: [10, 6, 4, 3, 0, 8, 7] Cost: 11.2 p: 0.9834312815364186
current Node: 756
sub node: [10, 6, 4, 3, 0, 8, 7, 9] Cost: 10.0 p: 0.9830899042203629
current Node: 766
sub node: [10, 6, 4, 3, 0, 8, 7, 9, 11] Cost: 8.8 p: 0.9827426954947567
current Node: 779
sub node: [10, 4, 3, 0, 8, 7, 9, 11] Cost: 11.2 p: 0.9823894341797499
current Node: 791
sub node: [10, 4, 3, 0, 7, 9, 11] Cost: 10.4 p: 0.9820298889021756
current Node: 805
sub node: [10, 4, 3, 0, 7, 9] Cost: 12.4 p: 0.9816638174772501
current Node: 807
sub node: [10, 4, 3, 0, 7, 9, 2] Cost: 13.8 p: 0.9812909662453775
current Node: 819
sub node: [10, 4, 3, 0, 7, 9, 2, 6] Cost: 11.4 p: 0.9809110693601925
current Node: 832
sub node: [10, 4, 3, 7, 9, 2, 6] Cost: 10.6 p: 0.9805238480235906
current Node: 846
sub node: [10, 4, 3, 9, 2, 6] Cost: 9.0 p: 0.9801290096630667
current Node: 857
sub node: [10, 3, 9, 2, 6] Cost: 7.2 p: 0.9797262470462058
current Node: 862
sub node: [10, 3, 9, 2, 6, 1] Cost: 6.4 p: 0.9793152373266383
current Node: 875
sub node: [10, 3, 9, 2, 6, 1, 5] Cost: 8.6 p: 0.978895641015177
current Node: 893
sub node: [3, 9, 2, 6, 1, 5] Cost: 8.4 p: 0.9784671008691852
current Node: 894
sub node: [3, 9, 2, 6, 1, 5, 0] Cost: 7.2 p: 0.9780292406924814
current Node: 907
sub node: [3, 9, 2, 6, 1, 5, 0, 8] Cost: 8.0 p: 0.9775816640372483
current Node: 924
sub node: [3, 9, 2, 1, 5, 0, 8] Cost: 6.4 p: 0.9771239527984675
current Node: 935
sub node: [3, 9, 2, 1, 0, 8] Cost: 7.8 p: 0.9766556656903498

current Node: 943
sub node: [3, 9, 2, 1, 8] Cost: 10.6 p: 0.9761763365930243
current Node: 954
sub node: [3, 9, 2, 1, 8, 10] Cost: 10.4 p: 0.9756854727564112
current Node: 961
sub node: [3, 9, 2, 1, 8, 10, 0] Cost: 8.4 p: 0.9751825528466642
current Node: 973
sub node: [3, 9, 2, 1, 8, 10, 0, 5] Cost: 8.6 p: 0.9746670248188473
current Node: 982
sub node: [3, 9, 2, 1, 8, 10, 0, 5, 4] Cost: 8.8 p: 0.9741383035975382
current Node: 993
sub node: [3, 9, 2, 1, 8, 10, 0, 5, 4, 7] Cost: 10.4 p: 0.9735957685448197
current Node: 1006
sub node: [3, 9, 1, 8, 10, 0, 5, 4, 7] Cost: 12.2 p: 0.973038760692573
current Node: 1012
sub node: [3, 9, 1, 8, 10, 0, 5, 4, 7, 2] Cost: 10.4 p: 0.9724665797130801
current Node: 1023
sub node: [3, 9, 1, 8, 10, 0, 5, 4, 7, 2, 6] Cost: 8.0 p: 0.9718784805986174
current Node: 1034
sub node: [3, 9, 1, 8, 10, 0, 5, 4, 7, 2, 6, 11] Cost: 6.800000000000001 p:
0.971273670016911
current Node: 1044
sub node: [3, 9, 1, 8, 10, 5, 4, 7, 2, 6, 11] Cost: 8.0 p: 0.9706513023049518
current Node: 1054
sub node: [3, 9, 1, 8, 10, 5, 4, 7, 2, 6, 11, 0] Cost: 6.800000000000001 p:
0.9700104750586231
current Node: 1067
sub node: [9, 1, 8, 10, 5, 4, 7, 2, 6, 11, 0] Cost: 8.0 p: 0.9693502242697928
current Node: 1079
sub node: [9, 1, 8, 10, 5, 7, 2, 6, 11, 0] Cost: 9.8 p: 0.9686695189557932
current Node: 1090
sub node: [9, 8, 10, 5, 7, 2, 6, 11, 0] Cost: 8.2 p: 0.967967255218435
current Node: 1106
sub node: [9, 8, 10, 5, 2, 6, 11, 0] Cost: 8.6 p: 0.9672422496606469
current Node: 1112
sub node: [9, 8, 10, 5, 2, 6, 11, 0, 3] Cost: 9.4 p: 0.9664932320783065
current Node: 1130
sub node: [9, 10, 5, 2, 6, 11, 0, 3] Cost: 8.6 p: 0.9657188373325155
current Node: 1140
sub node: [9, 10, 2, 6, 11, 0, 3] Cost: 6.4 p: 0.9649175962931757
current Node: 1145
sub node: [9, 10, 2, 6, 11, 0, 3, 1] Cost: 6.0 p: 0.9640879257278014
current Node: 1158

sub node: [9, 10, 2, 6, 11, 0, 3, 1, 5] Cost: 8.2 p: 0.9632281169895901
current Node: 1169
sub node: [9, 10, 2, 6, 11, 0, 3, 1, 5, 7] Cost: 9.8 p: 0.9623363233352442
current Node: 1185
sub node: [10, 2, 6, 11, 0, 3, 1, 5, 7] Cost: 11.0 p: 0.9614105456751673
current Node: 1198
sub node: [10, 2, 6, 0, 3, 1, 5, 7] Cost: 12.2 p: 0.9604486165255215
current Node: 1205
sub node: [10, 6, 0, 3, 1, 5, 7] Cost: 14.0 p: 0.9594481818921239
current Node: 1212
sub node: [10, 6, 0, 3, 1, 5, 7, 4] Cost: 12.2 p: 0.9584066807688645
current Node: 1233
sub node: [6, 0, 3, 1, 5, 7, 4] Cost: 10.0 p: 0.9573213218765259
current Node: 1244
sub node: [6, 0, 3, 1, 5, 4] Cost: 8.4 p: 0.9561890571993957
current Node: 1249
sub node: [6, 0, 3, 1, 5, 4, 10] Cost: 10.6 p: 0.955006551794156
current Node: 1266
sub node: [6, 0, 3, 1, 4, 10] Cost: 8.4 p: 0.9537701492447671
current Node: 1268
sub node: [6, 0, 3, 1, 4, 10, 2] Cost: 8.6 p: 0.9524758320140643
current Node: 1285
sub node: [6, 0, 3, 4, 10, 2] Cost: 11.0 p: 0.9511191757919584
current Node: 1292
sub node: [6, 0, 3, 4, 10, 2, 7] Cost: 12.6 p: 0.9496952967543435
current Node: 1304
sub node: [6, 0, 3, 4, 10, 2, 7, 9] Cost: 11.4 p: 0.9481987904168037
current Node: 1312
sub node: [6, 0, 3, 4, 10, 2, 7, 9, 1] Cost: 9.0 p: 0.9466236604810158
current Node: 1330
sub node: [6, 0, 3, 4, 10, 2, 9, 1] Cost: 7.4 p: 0.9449632357136915
current Node: 1340
sub node: [0, 3, 4, 10, 2, 9, 1] Cost: 7.8 p: 0.943210072447436
current Node: 1344
sub node: [0, 3, 4, 10, 2, 9, 1, 7] Cost: 11.4 p: 0.9413558397227654
current Node: 1361
sub node: [0, 4, 10, 2, 9, 1, 7] Cost: 10.6 p: 0.9393911833644196
current Node: 1373
sub node: [0, 10, 2, 9, 1, 7] Cost: 10.4 p: 0.9373055643542394
current Node: 1380
sub node: [0, 10, 2, 9, 1, 7, 8] Cost: 11.2 p: 0.9350870656611575
current Node: 1388
sub node: [0, 10, 2, 9, 1, 7, 8, 3] Cost: 12.0 p: 0.9327221601260388

current Node: 1407
sub node: [0, 10, 2, 1, 7, 8, 3] Cost: 11.2 p: 0.9301954299507723
current Node: 1418
sub node: [0, 10, 2, 1, 7, 3] Cost: 10.4 p: 0.9274892256341004
current Node: 1429
sub node: [0, 10, 2, 1, 3] Cost: 7.2 p: 0.9245832485878818
current Node: 1439
sub node: [0, 10, 2, 3] Cost: 10.8 p: 0.9214540368112507
current Node: 1445
sub node: [0, 10, 2, 3, 5] Cost: 10.6 p: 0.9180743263999378
current Node: 1463
sub node: [0, 10, 3, 5] Cost: 12.8 p: 0.914412252601763
current Node: 1472
sub node: [0, 10, 3, 5, 9] Cost: 12.0 p: 0.9104303415332333
current Node: 1479
sub node: [0, 10, 3, 5, 9, 4] Cost: 11.8 p: 0.906084225957335
current Node: 1489
sub node: [0, 10, 3, 5, 9, 4, 2] Cost: 10.4 p: 0.901320993280245
current Node: 1499
sub node: [0, 10, 3, 5, 9, 4, 2, 6] Cost: 10.0 p: 0.8960770374447742
current Node: 1513
sub node: [0, 10, 5, 9, 4, 2, 6] Cost: 9.2 p: 0.8902752328681656
current Node: 1525
sub node: [0, 10, 5, 9, 2, 6] Cost: 7.0 p: 0.8838211687150287
current Node: 1536
sub node: [0, 10, 5, 9, 6] Cost: 9.2 p: 0.8765980605117787
current Node: 1548
sub node: [10, 5, 9, 6] Cost: 8.0 p: 0.8684597682807385
current Node: 1554
sub node: [10, 5, 9, 6, 2] Cost: 7.8 p: 0.8592210532447118
current Node: 1569
sub node: [10, 5, 9, 6, 2, 8] Cost: 7.0 p: 0.848643724093507
current Node: 1578
sub node: [10, 5, 9, 6, 2, 8, 4] Cost: 9.2 p: 0.8364165249590011
current Node: 1587
sub node: [10, 5, 9, 6, 2, 8, 4, 3] Cost: 10.0 p: 0.8221252534784707
current Node: 1604
sub node: [5, 9, 6, 2, 8, 4, 3] Cost: 9.8 p: 0.8052071981190895
current Node: 1606
sub node: [5, 9, 6, 2, 8, 4, 3, 1] Cost: 9.4 p: 0.7848796247983224
current Node: 1617
sub node: [5, 9, 6, 2, 8, 4, 3, 1, 7] Cost: 9.0 p: 0.7600238517581545
current Node: 1636

sub node: [5, 6, 2, 8, 4, 3, 1, 7] Cost: 8.2 p: 0.7289905498863166
current Node: 1646
sub node: [5, 2, 8, 4, 3, 1, 7] Cost: 6.6 p: 0.6892602602433394
current Node: 1654
sub node: [5, 8, 4, 3, 1, 7] Cost: 6.4 p: 0.6368304153840199
current Node: 1671
sub node: [5, 4, 3, 1, 7] Cost: 8.4 p: 0.5650901140591805
current Node: 1673
sub node: [5, 4, 3, 1, 7, 2] Cost: 7.8 p: 0.46289464535120395
current Node: 1691
sub node: [5, 4, 1, 7, 2] Cost: 7.0 p: 0.3130937574877489
current Node: 1704
sub node: [4, 1, 7, 2] Cost: 9.2 p: 0.11125026260489224
current Node: 1710
sub node: [4, 1, 7, 2, 8] Cost: 8.4 p: 0.00033381238991847447
current Node: 1718
sub node: [4, 1, 7, 2, 8, 0] Cost: 6.4 p: 0.0
current Node: 1730
sub node: [4, 1, 7, 2, 8, 0, 5] Cost: 4.6 p: -1
current Node: -1
did not found the right Answer exiting Now ...
final Partitions: [4, 1, 7, 2, 0, 5] [3, 6, 8, 9, 10, 11] cost: 5.8

توجه شود که تابع نمایی سرعت رشد بیشتری دارد بنابراین بیشتر موقعیت تصادفی پیدا می کند و بیشتر طول میکشد تا به جواب برسد(اگر به جواب برسد).

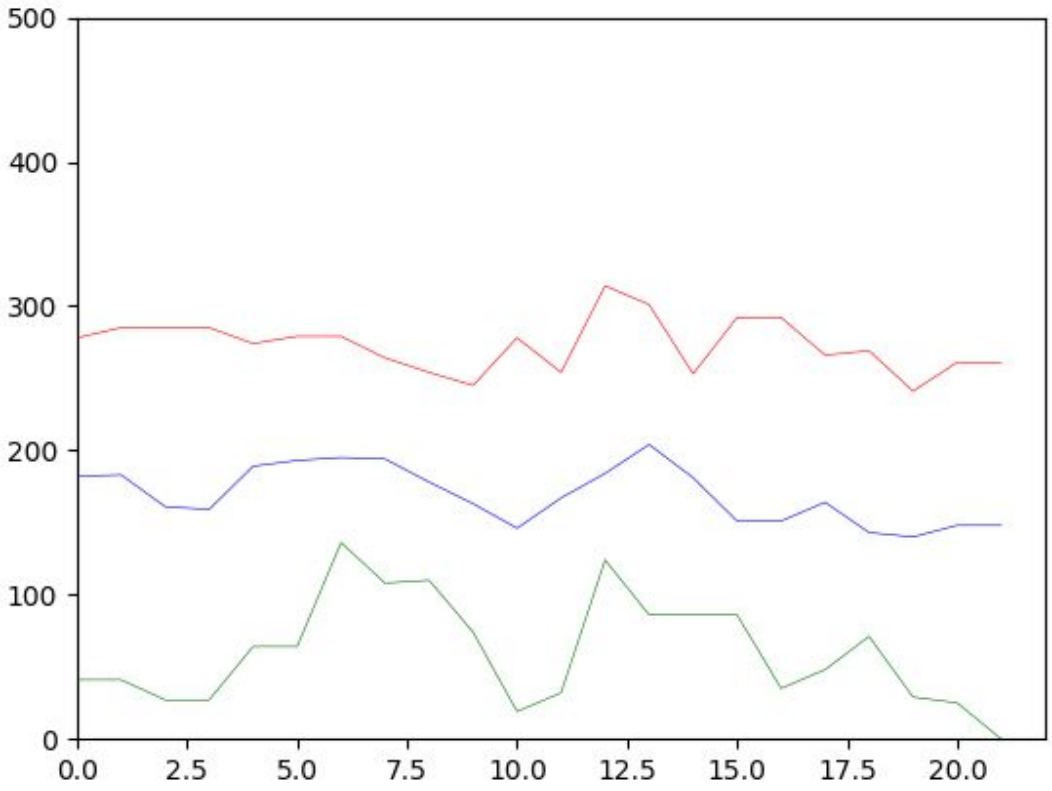
ژنتیک:

الف)

خروجی الگوریتم(pm=0.2,pc=0.25,chromosome_number =24):

Generation 1 : best fit: [27, 15, 8, 0] mid fit: [33, 22, 15, 25] worst fit: [2, 38, 36, 33]
Generation 2 : best fit: [27, 15, 8, 0] mid fit: [38, 37, 37, 0] worst fit: [9, 38, 36, 33]
Generation 3 : best fit: [20, 16, 5, 0] mid fit: [27, 39, 24, 6] worst fit: [9, 38, 36, 33]
Generation 4 : best fit: [20, 16, 5, 0] mid fit: [38, 37, 21, 6] worst fit: [9, 38, 36, 33]
Generation 5 : best fit: [27, 6, 7, 11] mid fit: [32, 19, 33, 15] worst fit: [23, 34, 37, 28]
Generation 6 : best fit: [27, 6, 7, 11] mid fit: [28, 5, 37, 21] worst fit: [38, 19, 29, 39]
Generation 7 : best fit: [32, 21, 26, 6] mid fit: [32, 8, 33, 22] worst fit: [38, 19, 29, 39]
Generation 8 : best fit: [4, 21, 26, 6] mid fit: [4, 17, 16, 37] worst fit: [32, 37, 34, 24]
Generation 9 : best fit: [21, 13, 21, 10] mid fit: [21, 25, 21, 21] worst fit: [33, 29, 37, 23]
Generation 10 : best fit: [33, 20, 7, 5] mid fit: [15, 16, 20, 24] worst fit: [33, 36, 28, 24]
Generation 11 : best fit: [1, 9, 12, 1] mid fit: [33, 20, 31, 5] worst fit: [27, 40, 21, 37]
Generation 12 : best fit: [8, 9, 14, 1] mid fit: [8, 6, 21, 31] worst fit: [21, 25, 21, 40]

Generation 13 : best fit: [19, 6, 23, 16] mid fit: [37, 13, 23, 23] worst fit: [32, 25, 40, 38]
Generation 14 : best fit: [26, 0, 28, 4] mid fit: [1, 25, 23, 31] worst fit: [19, 25, 40, 38]
Generation 15 : best fit: [26, 0, 28, 4] mid fit: [3, 25, 24, 24] worst fit: [19, 1, 40, 38]
Generation 16 : best fit: [26, 0, 28, 4] mid fit: [3, 22, 24, 18] worst fit: [21, 34, 33, 36]
Generation 17 : best fit: [24, 13, 3, 4] mid fit: [37, 22, 6, 23] worst fit: [21, 34, 33, 36]
Generation 18 : best fit: [9, 1, 23, 2] mid fit: [3, 22, 23, 22] worst fit: [35, 21, 23, 40]
Generation 19 : best fit: [12, 11, 23, 2] mid fit: [14, 6, 23, 22] worst fit: [35, 35, 20, 36]
Generation 20 : best fit: [9, 1, 6, 10] mid fit: [1, 11, 23, 22] worst fit: [15, 25, 36, 27]
Generation 21 : best fit: [33, 10, 0, 3] mid fit: [2, 6, 6, 39] worst fit: [2, 22, 33, 39]
Generation 22 : best fit: [8, 10, 0, 3] mid fit: [2, 6, 6, 39] worst fit: [2, 22, 33, 39]
final Values: [8, 10, 0, 3]



خروجی الگوریتم(chromosome_number=50,pm=0.5,pc=0.25):

Generation 1 : best fit: [13, 9, 16, 3] mid fit: [28, 22, 21, 21] worst fit: [20, 40, 38, 26]
Generation 2 : best fit: [4, 5, 11, 1] mid fit: [29, 5, 0, 33] worst fit: [33, 40, 38, 26]
Generation 3 : best fit: [14, 2, 10, 1] mid fit: [40, 11, 6, 29] worst fit: [25, 27, 26, 34]
Generation 4 : best fit: [14, 2, 1, 1] mid fit: [9, 21, 0, 40] worst fit: [32, 25, 23, 39]
Generation 5 : best fit: [3, 20, 3, 6] mid fit: [30, 29, 12, 26] worst fit: [14, 35, 36, 37]
Generation 6 : best fit: [11, 5, 7, 1] mid fit: [8, 35, 36, 9] worst fit: [0, 38, 38, 38]

Generation 7 : best fit: [7, 16, 1, 3] mid fit: [26, 34, 19, 9] worst fit: [36, 40, 26, 39]
Generation 8 : best fit: [11, 28, 14, 1] mid fit: [17, 4, 25, 26] worst fit: [23, 25, 40, 39]
Generation 9 : best fit: [11, 4, 9, 4] mid fit: [29, 32, 4, 20] worst fit: [21, 36, 39, 34]
Generation 10 : best fit: [12, 10, 6, 11] mid fit: [2, 28, 2, 34] worst fit: [21, 36, 39, 34]
Generation 11 : best fit: [5, 27, 5, 7] mid fit: [35, 22, 16, 25] worst fit: [18, 37, 37, 36]
Generation 12 : best fit: [4, 0, 21, 5] mid fit: [20, 34, 36, 4] worst fit: [17, 27, 40, 39]
Generation 13 : best fit: [11, 11, 10, 5] mid fit: [27, 14, 7, 28] worst fit: [9, 39, 37, 37]
Generation 14 : best fit: [24, 22, 1, 9] mid fit: [18, 21, 23, 19] worst fit: [36, 27, 40, 34]
Generation 15 : best fit: [10, 6, 6, 2] mid fit: [8, 6, 40, 13] worst fit: [39, 27, 40, 34]
Generation 16 : best fit: [7, 26, 15, 2] mid fit: [28, 22, 5, 27] worst fit: [39, 27, 40, 34]
Generation 17 : best fit: [0, 10, 22, 0] mid fit: [8, 27, 30, 14] worst fit: [13, 40, 33, 29]
Generation 18 : best fit: [15, 12, 15, 10] mid fit: [13, 40, 20, 20] worst fit: [29, 37, 31, 38]
Generation 19 : best fit: [39, 21, 2, 11] mid fit: [27, 39, 17, 12] worst fit: [24, 38, 31, 38]
Generation 20 : best fit: [23, 25, 6, 5] mid fit: [21, 4, 37, 23] worst fit: [40, 19, 38, 39]
Generation 21 : best fit: [5, 19, 7, 9] mid fit: [34, 23, 14, 17] worst fit: [37, 23, 37, 39]
Generation 22 : best fit: [6, 3, 25, 2] mid fit: [25, 22, 17, 30] worst fit: [37, 32, 36, 38]
Generation 23 : best fit: [21, 2, 19, 2] mid fit: [8, 8, 31, 20] worst fit: [28, 36, 38, 40]
Generation 24 : best fit: [7, 11, 37, 0] mid fit: [24, 9, 21, 34] worst fit: [4, 30, 40, 40]
Generation 25 : best fit: [19, 7, 6, 2] mid fit: [1, 10, 22, 32] worst fit: [22, 36, 40, 24]
Generation 26 : best fit: [10, 21, 7, 4] mid fit: [33, 28, 22, 16] worst fit: [23, 36, 40, 24]
Generation 27 : best fit: [8, 31, 7, 0] mid fit: [36, 9, 24, 18] worst fit: [33, 35, 38, 25]
Generation 28 : best fit: [26, 13, 5, 4] mid fit: [35, 7, 31, 18] worst fit: [24, 40, 25, 40]
Generation 29 : best fit: [28, 9, 12, 9] mid fit: [23, 8, 8, 33] worst fit: [30, 39, 38, 36]
Generation 30 : best fit: [9, 7, 5, 1] mid fit: [16, 10, 37, 8] worst fit: [0, 20, 39, 40]
Generation 31 : best fit: [22, 7, 5, 1] mid fit: [30, 1, 13, 38] worst fit: [35, 34, 39, 40]
Generation 32 : best fit: [17, 10, 9, 7] mid fit: [29, 8, 11, 33] worst fit: [34, 37, 37, 32]
Generation 33 : best fit: [19, 4, 2, 3] mid fit: [15, 14, 35, 20] worst fit: [34, 37, 37, 32]
Generation 34 : best fit: [19, 4, 2, 7] mid fit: [8, 6, 39, 12] worst fit: [24, 39, 34, 39]
Generation 35 : best fit: [19, 2, 10, 1] mid fit: [27, 20, 25, 20] worst fit: [20, 30, 39, 38]
Generation 36 : best fit: [21, 18, 2, 5] mid fit: [2, 19, 34, 17] worst fit: [20, 30, 39, 38]
Generation 37 : best fit: [3, 2, 10, 8] mid fit: [23, 7, 23, 27] worst fit: [20, 30, 39, 38]
Generation 38 : best fit: [29, 25, 3, 3] mid fit: [39, 8, 0, 38] worst fit: [3, 40, 31, 37]
Generation 39 : best fit: [16, 4, 17, 15] mid fit: [2, 18, 34, 15] worst fit: [30, 39, 35, 39]
Generation 40 : best fit: [3, 27, 14, 7] mid fit: [26, 37, 31, 6] worst fit: [29, 28, 35, 40]
Generation 41 : best fit: [3, 15, 0, 6] mid fit: [2, 24, 30, 18] worst fit: [38, 10, 34, 38]
Generation 42 : best fit: [0, 23, 5, 0] mid fit: [3, 37, 33, 8] worst fit: [37, 36, 26, 33]
Generation 43 : best fit: [33, 21, 12, 6] mid fit: [32, 35, 7, 24] worst fit: [38, 28, 33, 35]
Generation 44 : best fit: [4, 7, 6, 16] mid fit: [10, 9, 27, 26] worst fit: [39, 36, 37, 29]
Generation 45 : best fit: [7, 9, 6, 5] mid fit: [2, 16, 40, 8] worst fit: [17, 33, 33, 33]
Generation 46 : best fit: [15, 0, 1, 18] mid fit: [37, 20, 38, 5] worst fit: [35, 40, 39, 40]
Generation 47 : best fit: [26, 5, 3, 10] mid fit: [22, 15, 21, 12] worst fit: [35, 40, 27, 40]
Generation 48 : best fit: [5, 4, 10, 11] mid fit: [25, 25, 14, 21] worst fit: [14, 34, 21, 39]
Generation 49 : best fit: [0, 8, 4, 19] mid fit: [23, 24, 24, 19] worst fit: [37, 29, 37, 30]

Generation 50 : best fit: [14, 26, 4, 2] mid fit: [17, 2, 37, 21] worst fit: [26, 37, 40, 39]
Generation 51 : best fit: [2, 13, 4, 3] mid fit: [29, 30, 16, 22] worst fit: [37, 21, 40, 39]
Generation 52 : best fit: [27, 2, 1, 5] mid fit: [10, 18, 36, 10] worst fit: [35, 30, 39, 35]
Generation 53 : best fit: [4, 12, 0, 0] mid fit: [26, 11, 19, 24] worst fit: [26, 39, 40, 34]
Generation 54 : best fit: [28, 3, 3, 5] mid fit: [31, 24, 36, 7] worst fit: [15, 25, 38, 39]
Generation 55 : best fit: [21, 10, 1, 0] mid fit: [13, 11, 30, 16] worst fit: [11, 35, 27, 35]
Generation 56 : best fit: [15, 12, 4, 9] mid fit: [12, 13, 27, 28] worst fit: [24, 39, 25, 37]
Generation 57 : best fit: [11, 7, 0, 26] mid fit: [33, 7, 26, 23] worst fit: [10, 35, 22, 35]
Generation 58 : best fit: [0, 32, 10, 0] mid fit: [35, 6, 12, 21] worst fit: [32, 12, 35, 29]
Generation 59 : best fit: [4, 23, 0, 1] mid fit: [0, 8, 10, 33] worst fit: [2, 21, 35, 37]
Generation 60 : best fit: [4, 9, 5, 6] mid fit: [23, 27, 28, 11] worst fit: [10, 22, 37, 29]
Generation 61 : best fit: [24, 11, 17, 0] mid fit: [3, 34, 34, 3] worst fit: [35, 39, 39, 38]
Generation 62 : best fit: [5, 11, 11, 12] mid fit: [5, 21, 2, 36] worst fit: [28, 3, 38, 40]
Generation 63 : best fit: [13, 4, 1, 11] mid fit: [11, 20, 21, 22] worst fit: [24, 37, 35, 35]
Generation 64 : best fit: [4, 16, 6, 12] mid fit: [23, 13, 1, 35] worst fit: [24, 37, 35, 32]
Generation 65 : best fit: [15, 35, 4, 6] mid fit: [6, 16, 5, 33] worst fit: [21, 16, 34, 39]
Generation 66 : best fit: [0, 13, 1, 5] mid fit: [38, 17, 12, 19] worst fit: [37, 13, 38, 34]
Generation 67 : best fit: [11, 2, 10, 10] mid fit: [25, 5, 40, 11] worst fit: [37, 29, 40, 34]
Generation 68 : best fit: [23, 8, 21, 1] mid fit: [12, 29, 12, 31] worst fit: [33, 37, 36, 36]
Generation 69 : best fit: [25, 5, 3, 9] mid fit: [37, 31, 23, 6] worst fit: [31, 40, 37, 34]
Generation 70 : best fit: [16, 5, 16, 7] mid fit: [2, 32, 27, 13] worst fit: [31, 29, 38, 33]
Generation 71 : best fit: [19, 7, 15, 7] mid fit: [14, 38, 34, 0] worst fit: [22, 28, 37, 34]
Generation 72 : best fit: [14, 32, 2, 3] mid fit: [24, 31, 20, 11] worst fit: [24, 18, 35, 40]
Generation 73 : best fit: [1, 10, 2, 5] mid fit: [27, 12, 24, 21] worst fit: [37, 18, 35, 40]
Generation 74 : best fit: [11, 31, 2, 1] mid fit: [12, 26, 9, 30] worst fit: [27, 33, 35, 40]
Generation 75 : best fit: [10, 4, 22, 6] mid fit: [13, 32, 11, 26] worst fit: [27, 38, 32, 38]
Generation 76 : best fit: [28, 2, 7, 3] mid fit: [27, 38, 15, 19] worst fit: [11, 32, 29, 40]
Generation 77 : best fit: [22, 0, 5, 6] mid fit: [22, 0, 15, 40] worst fit: [9, 35, 32, 35]
Generation 78 : best fit: [34, 8, 6, 0] mid fit: [2, 37, 24, 12] worst fit: [33, 31, 39, 40]
Generation 79 : best fit: [16, 34, 6, 0] mid fit: [13, 25, 0, 37] worst fit: [27, 40, 39, 24]
Generation 80 : best fit: [25, 7, 4, 17] mid fit: [34, 7, 6, 35] worst fit: [31, 35, 38, 38]
Generation 81 : best fit: [36, 4, 7, 10] mid fit: [21, 26, 36, 12] worst fit: [36, 37, 32, 33]
Generation 82 : best fit: [22, 0, 3, 7] mid fit: [18, 37, 2, 37] worst fit: [39, 37, 32, 33]
Generation 83 : best fit: [5, 1, 22, 0] mid fit: [18, 7, 22, 30] worst fit: [21, 38, 32, 40]
Generation 84 : best fit: [7, 10, 9, 5] mid fit: [37, 1, 27, 27] worst fit: [24, 34, 38, 37]
Generation 85 : best fit: [37, 16, 9, 5] mid fit: [9, 21, 40, 14] worst fit: [21, 30, 32, 40]
Generation 86 : best fit: [17, 1, 5, 0] mid fit: [39, 5, 32, 10] worst fit: [31, 40, 40, 31]
Generation 87 : best fit: [10, 12, 7, 0] mid fit: [22, 8, 15, 27] worst fit: [13, 36, 40, 37]
Generation 88 : best fit: [16, 7, 14, 3] mid fit: [36, 4, 32, 6] worst fit: [13, 36, 40, 37]
Generation 89 : best fit: [22, 3, 1, 9] mid fit: [36, 33, 34, 1] worst fit: [14, 36, 37, 33]
Generation 90 : best fit: [9, 3, 11, 0] mid fit: [39, 28, 18, 9] worst fit: [37, 34, 34, 37]
Generation 91 : best fit: [23, 7, 5, 3] mid fit: [0, 4, 21, 27] worst fit: [37, 14, 34, 37]
Generation 92 : best fit: [23, 7, 5, 3] mid fit: [3, 22, 1, 39] worst fit: [31, 25, 34, 36]

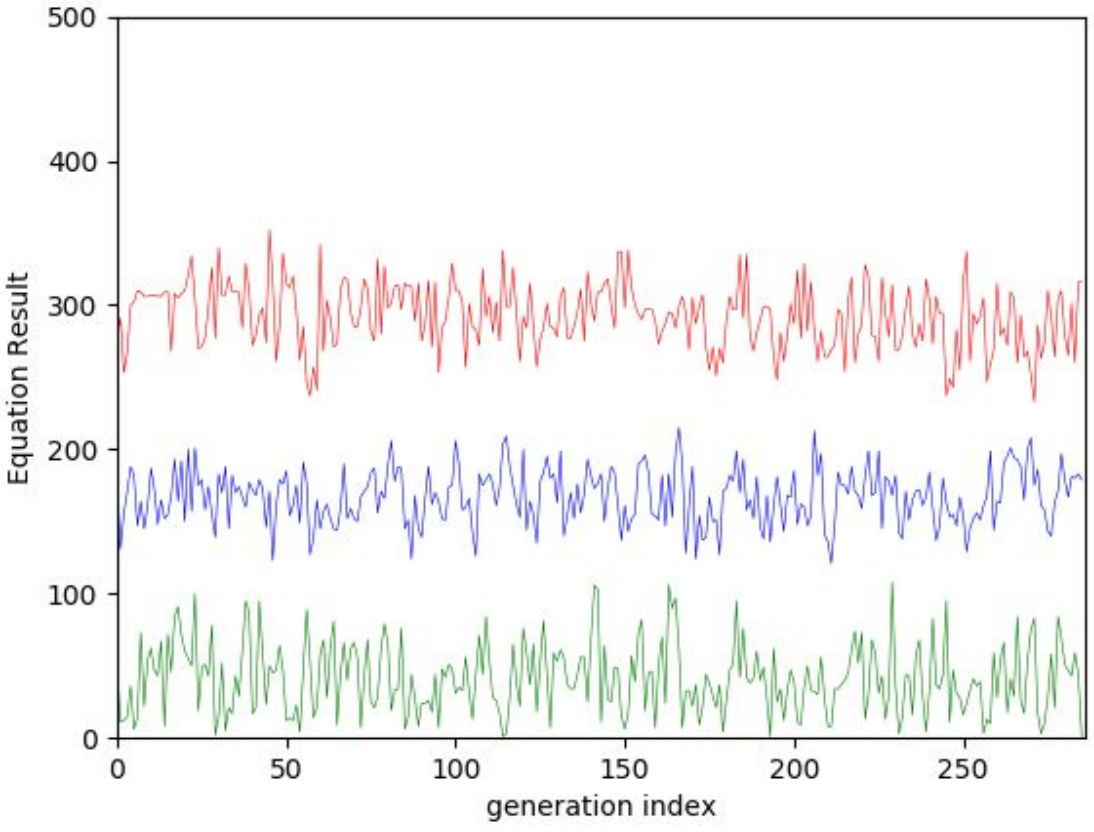
Generation 93 : best fit: [23, 8, 5, 3] mid fit: [32, 25, 34, 11] worst fit: [39, 36, 34, 36]
Generation 94 : best fit: [4, 14, 2, 5] mid fit: [16, 28, 6, 28] worst fit: [20, 39, 23, 36]
Generation 95 : best fit: [18, 8, 14, 2] mid fit: [18, 28, 32, 10] worst fit: [18, 40, 38, 36]
Generation 96 : best fit: [13, 8, 0, 1] mid fit: [10, 8, 22, 28] worst fit: [28, 20, 27, 36]
Generation 97 : best fit: [0, 2, 4, 18] mid fit: [35, 5, 17, 24] worst fit: [20, 22, 36, 38]
Generation 98 : best fit: [21, 5, 13, 3] mid fit: [31, 13, 26, 14] worst fit: [32, 15, 37, 39]
Generation 99 : best fit: [14, 17, 1, 10] mid fit: [34, 11, 14, 29] worst fit: [32, 29, 37, 36]
Generation 100 : best fit: [36, 3, 14, 1] mid fit: [5, 29, 24, 20] worst fit: [36, 37, 33, 40]
Generation 101 : best fit: [31, 20, 0, 0] mid fit: [21, 7, 21, 37] worst fit: [23, 27, 39, 39]
Generation 102 : best fit: [25, 3, 0, 11] mid fit: [6, 29, 2, 40] worst fit: [32, 28, 34, 40]
Generation 103 : best fit: [9, 2, 8, 9] mid fit: [15, 37, 3, 25] worst fit: [20, 37, 36, 35]
Generation 104 : best fit: [35, 7, 1, 11] mid fit: [31, 39, 25, 4] worst fit: [17, 26, 24, 39]
Generation 105 : best fit: [13, 3, 0, 13] mid fit: [0, 5, 40, 19] worst fit: [16, 38, 31, 39]
Generation 106 : best fit: [5, 6, 8, 3] mid fit: [16, 3, 19, 26] worst fit: [0, 38, 31, 39]
Generation 107 : best fit: [22, 27, 0, 2] mid fit: [27, 17, 31, 3] worst fit: [18, 38, 31, 34]
Generation 108 : best fit: [0, 13, 3, 19] mid fit: [7, 30, 24, 21] worst fit: [37, 23, 31, 34]
Generation 109 : best fit: [1, 7, 23, 0] mid fit: [9, 10, 34, 21] worst fit: [32, 37, 33, 40]
Generation 110 : best fit: [31, 3, 25, 3] mid fit: [7, 21, 6, 38] worst fit: [12, 39, 34, 35]
Generation 111 : best fit: [20, 1, 13, 8] mid fit: [21, 17, 20, 27] worst fit: [20, 40, 34, 36]
Generation 112 : best fit: [21, 11, 7, 1] mid fit: [37, 18, 21, 20] worst fit: [38, 26, 33, 33]
Generation 113 : best fit: [14, 10, 8, 2] mid fit: [27, 6, 14, 30] worst fit: [31, 40, 33, 33]
Generation 114 : best fit: [1, 16, 4, 3] mid fit: [13, 29, 18, 22] worst fit: [34, 27, 33, 32]
Generation 115 : best fit: [3, 2, 10, 1] mid fit: [23, 1, 37, 27] worst fit: [39, 40, 33, 40]
Generation 116 : best fit: [34, 0, 1, 0] mid fit: [1, 12, 28, 35] worst fit: [0, 40, 33, 40]
Generation 117 : best fit: [10, 7, 8, 5] mid fit: [12, 28, 16, 28] worst fit: [0, 40, 33, 40]
Generation 118 : best fit: [19, 14, 14, 4] mid fit: [24, 19, 23, 21] worst fit: [31, 40, 33, 39]
Generation 119 : best fit: [19, 14, 5, 4] mid fit: [29, 35, 25, 7] worst fit: [36, 19, 33, 40]
Generation 120 : best fit: [19, 6, 5, 4] mid fit: [28, 33, 1, 24] worst fit: [20, 33, 25, 35]
Generation 121 : best fit: [12, 3, 10, 17] mid fit: [27, 35, 33, 11] worst fit: [13, 27, 39, 37]
Generation 122 : best fit: [4, 1, 15, 12] mid fit: [12, 35, 10, 18] worst fit: [9, 35, 39, 32]
Generation 123 : best fit: [24, 1, 1, 12] mid fit: [33, 28, 33, 4] worst fit: [33, 33, 36, 37]
Generation 124 : best fit: [13, 36, 0, 5] mid fit: [22, 12, 11, 29] worst fit: [38, 31, 26, 37]
Generation 125 : best fit: [2, 1, 2, 12] mid fit: [24, 37, 23, 2] worst fit: [28, 32, 19, 37]
Generation 126 : best fit: [38, 17, 3, 3] mid fit: [11, 7, 23, 31] worst fit: [28, 36, 36, 27]
Generation 127 : best fit: [37, 16, 4, 10] mid fit: [40, 10, 12, 32] worst fit: [36, 36, 36, 27]
Generation 128 : best fit: [34, 16, 6, 0] mid fit: [22, 10, 27, 28] worst fit: [24, 37, 37, 33]
Generation 129 : best fit: [0, 5, 3, 7] mid fit: [35, 40, 7, 21] worst fit: [16, 31, 33, 37]
Generation 130 : best fit: [34, 0, 7, 11] mid fit: [40, 14, 21, 23] worst fit: [26, 36, 30, 34]
Generation 131 : best fit: [23, 19, 4, 5] mid fit: [39, 0, 16, 29] worst fit: [29, 22, 31, 38]
Generation 132 : best fit: [21, 5, 14, 7] mid fit: [13, 36, 6, 34] worst fit: [21, 40, 31, 38]
Generation 133 : best fit: [19, 25, 1, 6] mid fit: [32, 19, 26, 8] worst fit: [17, 40, 37, 36]
Generation 134 : best fit: [1, 11, 6, 9] mid fit: [9, 10, 27, 21] worst fit: [22, 40, 21, 38]
Generation 135 : best fit: [3, 2, 13, 7] mid fit: [29, 29, 9, 24] worst fit: [22, 40, 21, 38]

Generation 136 : best fit: [4, 1, 4, 14] mid fit: [14, 37, 4, 23] worst fit: [19, 28, 34, 37]
Generation 137 : best fit: [11, 1, 1, 17] mid fit: [24, 4, 8, 40] worst fit: [33, 37, 36, 30]
Generation 138 : best fit: [18, 16, 10, 4] mid fit: [36, 38, 16, 9] worst fit: [37, 38, 26, 40]
Generation 139 : best fit: [25, 0, 9, 11] mid fit: [31, 33, 28, 7] worst fit: [37, 18, 30, 38]
Generation 140 : best fit: [14, 24, 1, 0] mid fit: [27, 15, 32, 20] worst fit: [36, 32, 37, 38]
Generation 141 : best fit: [14, 24, 14, 1] mid fit: [10, 5, 14, 40] worst fit: [39, 23, 32, 40]
Generation 142 : best fit: [10, 38, 4, 12] mid fit: [21, 3, 38, 18] worst fit: [27, 38, 30, 34]
Generation 143 : best fit: [25, 20, 22, 3] mid fit: [12, 12, 16, 33] worst fit: [24, 35, 37, 36]
Generation 144 : best fit: [1, 9, 0, 8] mid fit: [30, 30, 19, 19] worst fit: [24, 35, 37, 36]
Generation 145 : best fit: [16, 22, 8, 5] mid fit: [5, 32, 19, 16] worst fit: [31, 33, 38, 36]
Generation 146 : best fit: [3, 2, 0, 15] mid fit: [1, 20, 32, 19] worst fit: [29, 29, 37, 40]
Generation 147 : best fit: [29, 13, 2, 1] mid fit: [0, 3, 30, 33] worst fit: [9, 32, 37, 39]
Generation 148 : best fit: [39, 0, 14, 2] mid fit: [6, 27, 40, 11] worst fit: [20, 38, 24, 39]
Generation 149 : best fit: [18, 5, 0, 15] mid fit: [17, 17, 18, 24] worst fit: [36, 39, 37, 38]
Generation 150 : best fit: [8, 5, 1, 1] mid fit: [5, 36, 32, 1] worst fit: [36, 39, 37, 38]
Generation 151 : best fit: [6, 17, 2, 0] mid fit: [33, 22, 40, 1] worst fit: [21, 26, 39, 37]
Generation 152 : best fit: [37, 0, 0, 5] mid fit: [30, 20, 15, 17] worst fit: [36, 31, 40, 40]
Generation 153 : best fit: [35, 16, 10, 0] mid fit: [38, 35, 4, 18] worst fit: [38, 17, 40, 40]
Generation 154 : best fit: [2, 1, 13, 9] mid fit: [21, 35, 4, 23] worst fit: [22, 28, 38, 37]
Generation 155 : best fit: [7, 11, 13, 11] mid fit: [35, 6, 9, 39] worst fit: [21, 20, 39, 39]
Generation 156 : best fit: [24, 7, 16, 9] mid fit: [33, 19, 31, 17] worst fit: [28, 19, 40, 36]
Generation 157 : best fit: [27, 1, 2, 6] mid fit: [32, 25, 30, 16] worst fit: [25, 27, 34, 39]
Generation 158 : best fit: [7, 16, 2, 10] mid fit: [26, 22, 30, 16] worst fit: [25, 27, 34, 39]
Generation 159 : best fit: [9, 1, 25, 0] mid fit: [19, 10, 8, 33] worst fit: [25, 27, 34, 39]
Generation 160 : best fit: [14, 3, 1, 6] mid fit: [30, 15, 26, 14] worst fit: [40, 36, 27, 34]
Generation 161 : best fit: [11, 19, 14, 0] mid fit: [30, 15, 25, 14] worst fit: [40, 40, 11, 40]
Generation 162 : best fit: [7, 25, 3, 11] mid fit: [14, 33, 2, 34] worst fit: [11, 32, 29, 40]
Generation 163 : best fit: [15, 10, 7, 2] mid fit: [7, 10, 40, 10] worst fit: [16, 32, 29, 40]
Generation 164 : best fit: [7, 10, 37, 2] mid fit: [7, 23, 14, 32] worst fit: [7, 30, 40, 37]
Generation 165 : best fit: [7, 7, 15, 16] mid fit: [37, 18, 28, 9] worst fit: [14, 38, 28, 40]
Generation 166 : best fit: [10, 26, 5, 15] mid fit: [9, 15, 25, 31] worst fit: [35, 23, 33, 36]
Generation 167 : best fit: [19, 15, 3, 11] mid fit: [10, 20, 39, 22] worst fit: [27, 32, 40, 32]
Generation 168 : best fit: [6, 9, 5, 2] mid fit: [19, 15, 11, 37] worst fit: [35, 37, 39, 30]
Generation 169 : best fit: [8, 1, 10, 8] mid fit: [4, 11, 30, 13] worst fit: [40, 37, 23, 38]
Generation 170 : best fit: [8, 12, 3, 8] mid fit: [8, 20, 8, 32] worst fit: [35, 24, 38, 28]
Generation 171 : best fit: [4, 7, 8, 5] mid fit: [23, 15, 17, 31] worst fit: [38, 36, 33, 34]
Generation 172 : best fit: [0, 28, 3, 3] mid fit: [14, 10, 22, 16] worst fit: [26, 27, 33, 37]
Generation 173 : best fit: [2, 0, 5, 8] mid fit: [7, 3, 15, 34] worst fit: [26, 33, 33, 37]
Generation 174 : best fit: [0, 8, 6, 7] mid fit: [6, 17, 19, 20] worst fit: [28, 23, 39, 39]
Generation 175 : best fit: [22, 8, 6, 7] mid fit: [24, 1, 11, 30] worst fit: [30, 32, 27, 34]
Generation 176 : best fit: [8, 11, 4, 9] mid fit: [7, 11, 10, 37] worst fit: [29, 29, 24, 34]
Generation 177 : best fit: [9, 17, 6, 2] mid fit: [3, 5, 23, 27] worst fit: [15, 37, 30, 33]
Generation 178 : best fit: [4, 2, 15, 2] mid fit: [8, 30, 5, 27] worst fit: [27, 37, 10, 40]

Generation 179 : best fit: [9, 25, 2, 3] mid fit: [4, 15, 15, 22] worst fit: [36, 14, 30, 39]
Generation 180 : best fit: [22, 7, 0, 2] mid fit: [27, 2, 28, 24] worst fit: [12, 33, 26, 36]
Generation 181 : best fit: [3, 15, 7, 3] mid fit: [29, 9, 22, 25] worst fit: [30, 25, 35, 36]
Generation 182 : best fit: [38, 0, 3, 10] mid fit: [3, 25, 19, 28] worst fit: [27, 29, 35, 39]
Generation 183 : best fit: [7, 14, 4, 11] mid fit: [19, 33, 39, 4] worst fit: [40, 22, 35, 37]
Generation 184 : best fit: [33, 10, 2, 19] mid fit: [32, 9, 27, 27] worst fit: [32, 32, 27, 40]
Generation 185 : best fit: [18, 5, 6, 9] mid fit: [4, 2, 14, 40] worst fit: [38, 35, 37, 39]
Generation 186 : best fit: [37, 5, 23, 0] mid fit: [9, 36, 36, 11] worst fit: [21, 22, 38, 38]
Generation 187 : best fit: [10, 4, 12, 7] mid fit: [27, 11, 31, 14] worst fit: [34, 37, 37, 39]
Generation 188 : best fit: [15, 19, 7, 1] mid fit: [34, 0, 33, 18] worst fit: [6, 31, 34, 37]
Generation 189 : best fit: [16, 14, 7, 4] mid fit: [40, 5, 2, 34] worst fit: [22, 18, 34, 37]
Generation 190 : best fit: [40, 5, 2, 11] mid fit: [30, 1, 35, 21] worst fit: [15, 39, 40, 27]
Generation 191 : best fit: [12, 20, 4, 4] mid fit: [0, 5, 8, 38] worst fit: [36, 19, 36, 37]
Generation 192 : best fit: [22, 9, 3, 10] mid fit: [40, 0, 2, 33] worst fit: [11, 36, 36, 37]
Generation 193 : best fit: [2, 13, 13, 0] mid fit: [9, 3, 39, 19] worst fit: [11, 36, 36, 37]
Generation 194 : best fit: [14, 11, 1, 0] mid fit: [39, 25, 21, 6] worst fit: [11, 35, 36, 37]
Generation 195 : best fit: [0, 3, 32, 0] mid fit: [37, 6, 36, 11] worst fit: [38, 10, 39, 32]
Generation 196 : best fit: [5, 19, 6, 1] mid fit: [19, 17, 12, 33] worst fit: [39, 31, 25, 28]
Generation 197 : best fit: [6, 6, 4, 11] mid fit: [19, 17, 27, 17] worst fit: [21, 31, 38, 31]
Generation 198 : best fit: [7, 2, 7, 5] mid fit: [18, 26, 23, 11] worst fit: [27, 13, 32, 38]
Generation 199 : best fit: [7, 2, 14, 5] mid fit: [40, 4, 24, 22] worst fit: [39, 4, 37, 40]
Generation 200 : best fit: [4, 11, 7, 10] mid fit: [32, 18, 17, 22] worst fit: [11, 27, 40, 39]
Generation 201 : best fit: [20, 22, 1, 3] mid fit: [6, 33, 7, 33] worst fit: [30, 28, 27, 40]
Generation 202 : best fit: [11, 20, 0, 0] mid fit: [3, 27, 17, 20] worst fit: [40, 28, 36, 40]
Generation 203 : best fit: [1, 1, 6, 7] mid fit: [17, 13, 17, 27] worst fit: [14, 28, 29, 40]
Generation 204 : best fit: [1, 14, 6, 7] mid fit: [35, 34, 15, 13] worst fit: [39, 34, 34, 40]
Generation 205 : best fit: [3, 2, 5, 17] mid fit: [5, 32, 10, 22] worst fit: [33, 36, 19, 40]
Generation 206 : best fit: [30, 15, 3, 1] mid fit: [9, 17, 5, 34] worst fit: [35, 38, 35, 35]
Generation 207 : best fit: [11, 17, 5, 3] mid fit: [34, 1, 39, 25] worst fit: [10, 32, 31, 39]
Generation 208 : best fit: [36, 6, 2, 4] mid fit: [21, 24, 35, 12] worst fit: [8, 36, 35, 29]
Generation 209 : best fit: [38, 10, 2, 8] mid fit: [10, 3, 35, 29] worst fit: [21, 29, 29, 39]
Generation 210 : best fit: [13, 12, 5, 4] mid fit: [18, 35, 15, 12] worst fit: [11, 33, 25, 38]
Generation 211 : best fit: [2, 10, 6, 2] mid fit: [18, 20, 38, 1] worst fit: [11, 37, 21, 39]
Generation 212 : best fit: [2, 10, 6, 2] mid fit: [33, 4, 32, 6] worst fit: [30, 39, 35, 24]
Generation 213 : best fit: [21, 11, 9, 1] mid fit: [11, 23, 34, 7] worst fit: [34, 30, 22, 38]
Generation 214 : best fit: [21, 11, 9, 1] mid fit: [37, 8, 17, 30] worst fit: [21, 40, 40, 29]
Generation 215 : best fit: [18, 26, 1, 1] mid fit: [37, 7, 24, 23] worst fit: [11, 30, 37, 38]
Generation 216 : best fit: [34, 4, 10, 2] mid fit: [12, 31, 11, 27] worst fit: [34, 9, 38, 32]
Generation 217 : best fit: [26, 8, 9, 4] mid fit: [22, 23, 39, 6] worst fit: [36, 24, 36, 38]
Generation 218 : best fit: [39, 31, 0, 0] mid fit: [22, 8, 38, 19] worst fit: [36, 39, 35, 35]
Generation 219 : best fit: [34, 22, 8, 3] mid fit: [23, 33, 26, 11] worst fit: [22, 8, 34, 40]
Generation 220 : best fit: [34, 8, 14, 0] mid fit: [40, 35, 10, 17] worst fit: [39, 36, 39, 23]
Generation 221 : best fit: [6, 2, 25, 7] mid fit: [6, 2, 23, 30] worst fit: [33, 19, 33, 39]

Generation 222 : best fit: [8, 6, 3, 6] mid fit: [35, 21, 20, 21] worst fit: [39, 36, 35, 38]
Generation 223 : best fit: [20, 22, 2, 0] mid fit: [1, 26, 30, 24] worst fit: [39, 26, 39, 38]
Generation 224 : best fit: [29, 15, 11, 4] mid fit: [30, 16, 14, 26] worst fit: [21, 37, 32, 32]
Generation 225 : best fit: [12, 36, 1, 3] mid fit: [12, 23, 36, 11] worst fit: [36, 23, 36, 32]
Generation 226 : best fit: [15, 17, 2, 4] mid fit: [1, 30, 26, 25] worst fit: [23, 1, 40, 39]
Generation 227 : best fit: [0, 34, 1, 3] mid fit: [39, 13, 24, 12] worst fit: [34, 40, 35, 35]
Generation 228 : best fit: [12, 8, 7, 1] mid fit: [7, 19, 40, 14] worst fit: [28, 29, 32, 38]
Generation 229 : best fit: [11, 21, 0, 4] mid fit: [22, 2, 40, 18] worst fit: [21, 22, 31, 40]
Generation 230 : best fit: [0, 2, 8, 30] mid fit: [38, 25, 16, 18] worst fit: [17, 36, 35, 40]
Generation 231 : best fit: [33, 2, 5, 6] mid fit: [36, 16, 26, 19] worst fit: [22, 23, 35, 34]
Generation 232 : best fit: [0, 2, 11, 0] mid fit: [7, 8, 37, 12] worst fit: [22, 23, 35, 34]
Generation 233 : best fit: [23, 2, 5, 3] mid fit: [37, 16, 3, 25] worst fit: [39, 9, 39, 36]
Generation 234 : best fit: [16, 5, 14, 4] mid fit: [22, 13, 21, 25] worst fit: [13, 33, 33, 40]
Generation 235 : best fit: [9, 12, 14, 2] mid fit: [29, 8, 26, 17] worst fit: [34, 33, 31, 40]
Generation 236 : best fit: [27, 0, 3, 5] mid fit: [29, 11, 14, 28] worst fit: [14, 30, 38, 33]
Generation 237 : best fit: [13, 2, 20, 5] mid fit: [4, 6, 21, 33] worst fit: [12, 38, 21, 40]
Generation 238 : best fit: [3, 23, 9, 8] mid fit: [2, 31, 4, 34] worst fit: [29, 35, 40, 27]
Generation 239 : best fit: [12, 17, 4, 3] mid fit: [33, 11, 22, 20] worst fit: [27, 39, 18, 39]
Generation 240 : best fit: [9, 0, 5, 5] mid fit: [12, 21, 19, 24] worst fit: [33, 39, 29, 40]
Generation 241 : best fit: [18, 17, 4, 2] mid fit: [10, 10, 18, 35] worst fit: [2, 34, 38, 40]
Generation 242 : best fit: [1, 18, 6, 17] mid fit: [27, 25, 34, 5] worst fit: [28, 31, 29, 34]
Generation 243 : best fit: [18, 16, 9, 0] mid fit: [18, 12, 5, 30] worst fit: [17, 30, 39, 38]
Generation 244 : best fit: [7, 8, 5, 9] mid fit: [39, 18, 37, 0] worst fit: [32, 27, 36, 35]
Generation 245 : best fit: [1, 8, 10, 9] mid fit: [17, 30, 25, 17] worst fit: [15, 27, 39, 37]
Generation 246 : best fit: [7, 4, 40, 0] mid fit: [31, 31, 22, 11] worst fit: [32, 11, 21, 40]
Generation 247 : best fit: [16, 6, 1, 5] mid fit: [19, 7, 20, 25] worst fit: [17, 18, 40, 29]
Generation 248 : best fit: [28, 3, 3, 11] mid fit: [20, 12, 25, 19] worst fit: [27, 16, 40, 26]
Generation 249 : best fit: [18, 5, 1, 10] mid fit: [16, 37, 11, 16] worst fit: [35, 20, 32, 38]
Generation 250 : best fit: [18, 5, 12, 1] mid fit: [4, 21, 31, 17] worst fit: [25, 20, 30, 35]
Generation 251 : best fit: [6, 5, 12, 1] mid fit: [16, 0, 18, 31] worst fit: [29, 30, 36, 40]
Generation 252 : best fit: [13, 0, 9, 6] mid fit: [4, 2, 31, 17] worst fit: [29, 40, 36, 40]
Generation 253 : best fit: [18, 0, 9, 7] mid fit: [4, 35, 37, 0] worst fit: [32, 8, 39, 34]
Generation 254 : best fit: [35, 20, 2, 0] mid fit: [19, 20, 11, 25] worst fit: [29, 31, 33, 36]
Generation 255 : best fit: [33, 5, 7, 3] mid fit: [3, 35, 9, 24] worst fit: [6, 35, 37, 35]
Generation 256 : best fit: [3, 26, 8, 0] mid fit: [15, 31, 38, 0] worst fit: [22, 28, 38, 35]
Generation 257 : best fit: [21, 5, 4, 0] mid fit: [26, 11, 19, 24] worst fit: [25, 38, 40, 31]
Generation 258 : best fit: [2, 24, 1, 0] mid fit: [27, 37, 10, 19] worst fit: [12, 38, 37, 22]
Generation 259 : best fit: [26, 0, 4, 3] mid fit: [28, 4, 29, 29] worst fit: [25, 14, 34, 36]
Generation 260 : best fit: [37, 7, 0, 12] mid fit: [38, 38, 10, 10] worst fit: [27, 28, 26, 37]
Generation 261 : best fit: [17, 10, 1, 7] mid fit: [39, 26, 3, 26] worst fit: [37, 25, 36, 40]
Generation 262 : best fit: [12, 14, 1, 13] mid fit: [34, 25, 9, 23] worst fit: [37, 29, 32, 32]
Generation 263 : best fit: [2, 3, 14, 14] mid fit: [37, 26, 10, 28] worst fit: [37, 32, 25, 37]
Generation 264 : best fit: [2, 3, 14, 2] mid fit: [30, 4, 31, 26] worst fit: [16, 35, 38, 28]

Generation 265 : best fit: [35, 1, 4, 8] mid fit: [16, 20, 39, 17] worst fit: [16, 40, 35, 37]
Generation 266 : best fit: [6, 10, 13, 1] mid fit: [29, 23, 9, 33] worst fit: [18, 37, 35, 37]
Generation 267 : best fit: [6, 32, 14, 3] mid fit: [35, 7, 16, 34] worst fit: [8, 40, 36, 26]
Generation 268 : best fit: [6, 24, 0, 3] mid fit: [38, 22, 10, 29] worst fit: [34, 11, 39, 40]
Generation 269 : best fit: [0, 21, 5, 0] mid fit: [17, 11, 30, 19] worst fit: [33, 22, 37, 29]
Generation 270 : best fit: [26, 11, 6, 7] mid fit: [4, 36, 36, 14] worst fit: [17, 37, 39, 25]
Generation 271 : best fit: [9, 39, 0, 7] mid fit: [8, 39, 30, 18] worst fit: [33, 34, 13, 38]
Generation 272 : best fit: [3, 2, 12, 20] mid fit: [39, 7, 2, 39] worst fit: [25, 9, 26, 38]
Generation 273 : best fit: [39, 4, 2, 1] mid fit: [16, 31, 4, 34] worst fit: [1, 32, 35, 39]
Generation 274 : best fit: [1, 15, 4, 0] mid fit: [32, 4, 3, 38] worst fit: [22, 11, 33, 40]
Generation 275 : best fit: [16, 2, 2, 6] mid fit: [22, 27, 14, 20] worst fit: [1, 33, 37, 34]
Generation 276 : best fit: [1, 4, 0, 15] mid fit: [11, 15, 6, 31] worst fit: [2, 36, 40, 39]
Generation 277 : best fit: [8, 33, 8, 0] mid fit: [34, 40, 18, 3] worst fit: [37, 36, 21, 40]
Generation 278 : best fit: [26, 1, 7, 3] mid fit: [24, 28, 21, 15] worst fit: [11, 35, 25, 37]
Generation 279 : best fit: [20, 12, 0, 20] mid fit: [15, 11, 10, 36] worst fit: [6, 38, 38, 37]
Generation 280 : best fit: [18, 1, 7, 17] mid fit: [28, 33, 9, 29] worst fit: [12, 38, 38, 37]
Generation 281 : best fit: [4, 33, 7, 0] mid fit: [24, 2, 21, 32] worst fit: [33, 21, 36, 34]
Generation 282 : best fit: [2, 13, 5, 11] mid fit: [27, 25, 34, 7] worst fit: [19, 21, 36, 34]
Generation 283 : best fit: [4, 14, 9, 6] mid fit: [8, 17, 25, 26] worst fit: [39, 31, 40, 30]
Generation 284 : best fit: [8, 23, 7, 6] mid fit: [27, 23, 21, 21] worst fit: [39, 28, 19, 37]
Generation 285 : best fit: [3, 0, 27, 0] mid fit: [26, 10, 35, 18] worst fit: [35, 28, 35, 40]
Generation 286 : best fit: [6, 2, 2, 6] mid fit: [37, 24, 14, 23] worst fit: [35, 28, 35, 40]
final Values: [6, 2, 2, 6]



ب)

اگر مقدار جهش به صفر میل کند الگوریتم در حلقه گیر می کند چرا که بعد از مدتی تمام کروموزم ها مقدار یکسان میگیرند و cross over کردن آنها بی فایده می شود.
اگر مقدار جهش به یک میل کند نیز الگوریتم ژنتیک بیهوده است چرا که تمام اعداد دوباره به صورت رندم تولید می شوند بنابراین هیچ یک از دو حالت بالا نتیجه خوبی نمی دهد.
مقدار مناسب جهش را میتوان بین 0.2 تا 0.4 تخمین زد گرچه ممکن است در برخی حالات مقادیر بیشتر بهتر عمل کنند.

ج)

افزایش جمعیت اگر چه باعث میشود که افراد با شایستگی های بهتری ظاهر شوند اما تا حدی مناسب است که محاسبات ناشی از جهش برای هر نسل زمان بیش از حد نبرد. اگر جمعیت خیلی کم باشد نیز مقدار جهش بسیار زیاد می شود و آن نیز مناسب نیست بهتر است مقدار جمعیت در حد متعادلی باشد تا بتوان به بهترین نتیجه ممکن رسید.

د)

با جمعیت 2: بعد از 14129 نسل
با جمعیت 20: بعد از 531 نسل
با جمعیت 200: بعد از 103 نسل
جمعیت 2000: بعد از 1 نسل (زمانبر)
مشکل دیگری که در افزایش بیش از حد نسل بوجود می آید این است که ممکن است نسل آنقدر زیاد شود که دیگر نتوان شایستگی را مورد ارزیابی قرار داد.