

مسئله اول

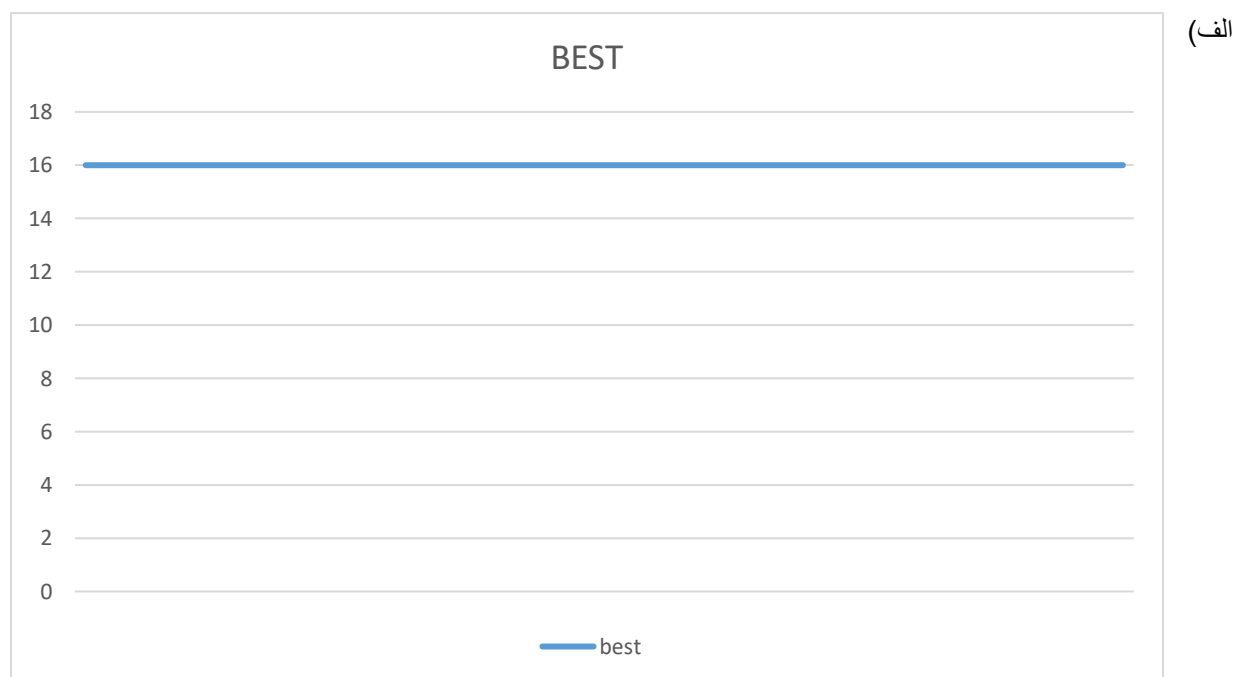
State ها یک آرایه است به اندازه ی تعداد node های گراف ما که در هر خانه از آن می گوید که این نود الان چه رنگی است. در هر مرحله action هایی که می تواند انجام دهد این است که کدام یک از نود ها را می تواند رنگش را عوض کند یعنی همه ی نود ها ی ممکن.

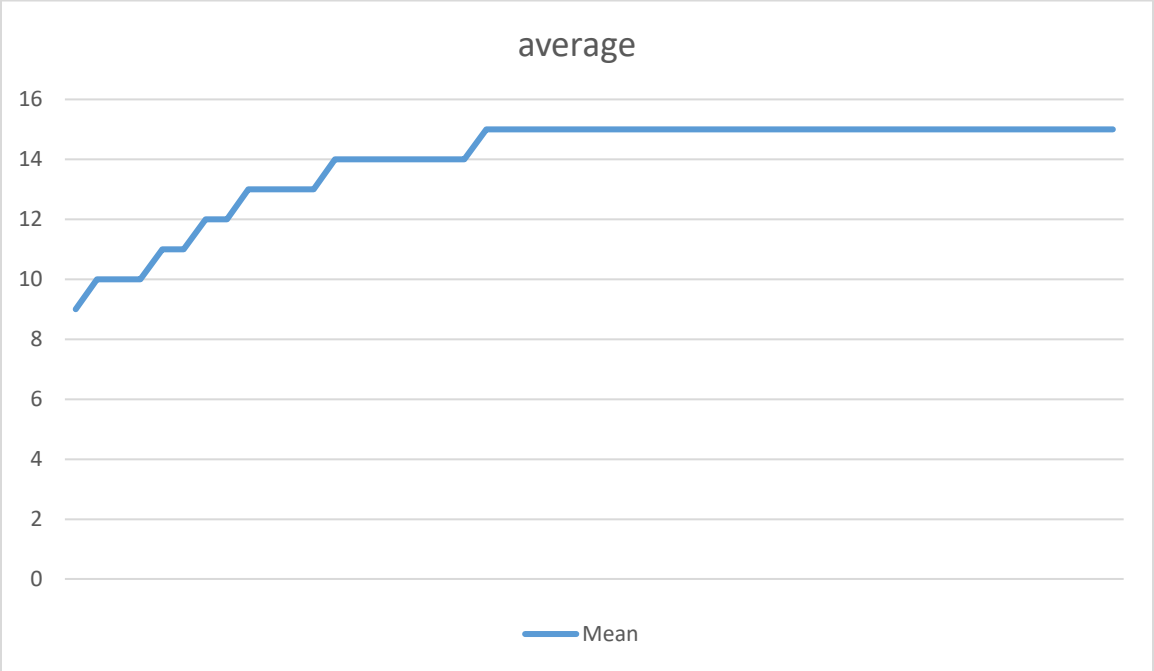
Goal هم نداریم. در هر مرحله تعداد نود های مجاور و هم رنگ رو می شمارد و سعی دارد تعداد آن ها را کم کند.

مسئله دوم

State ها یک آرایه است که از کنار هم قرار گرفتن سطرهای جدول به درست می آید. در هر مرحله برای action ها دو خانه ی رندم انتخاب می کنیم و مکانشون رو عوض می کنیم شماره ی خانه ی اول و شماره دوم که دو رقمی در نظر گرفته شده. برای به دست آوردن ارزش هر state در دیکشنری for می زنیم و حرف اول رو می یابیم حرف دوم هم اگر بالا پایین چپ یا راست حرف اول بود تا انتها ی کلمه پیش می رویم. اگر تونستیم یک روند پیدا کنیم به سراغ کلمه ی دیگر می رویم. هدف این است که ماکزیمم کلمات پیدا شوند یعنی تعداد کل کلمات دیکشنری منهای تعداد کل پیدا شده را کمینه کنیم.

مسئله سوم







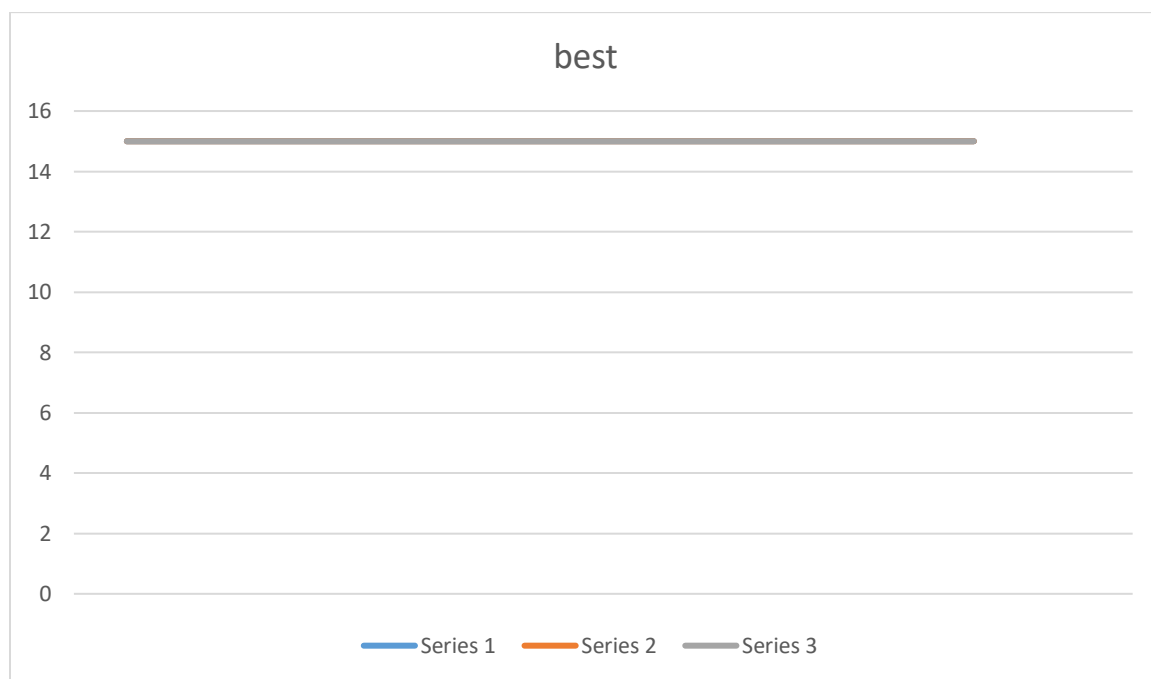
(ب)

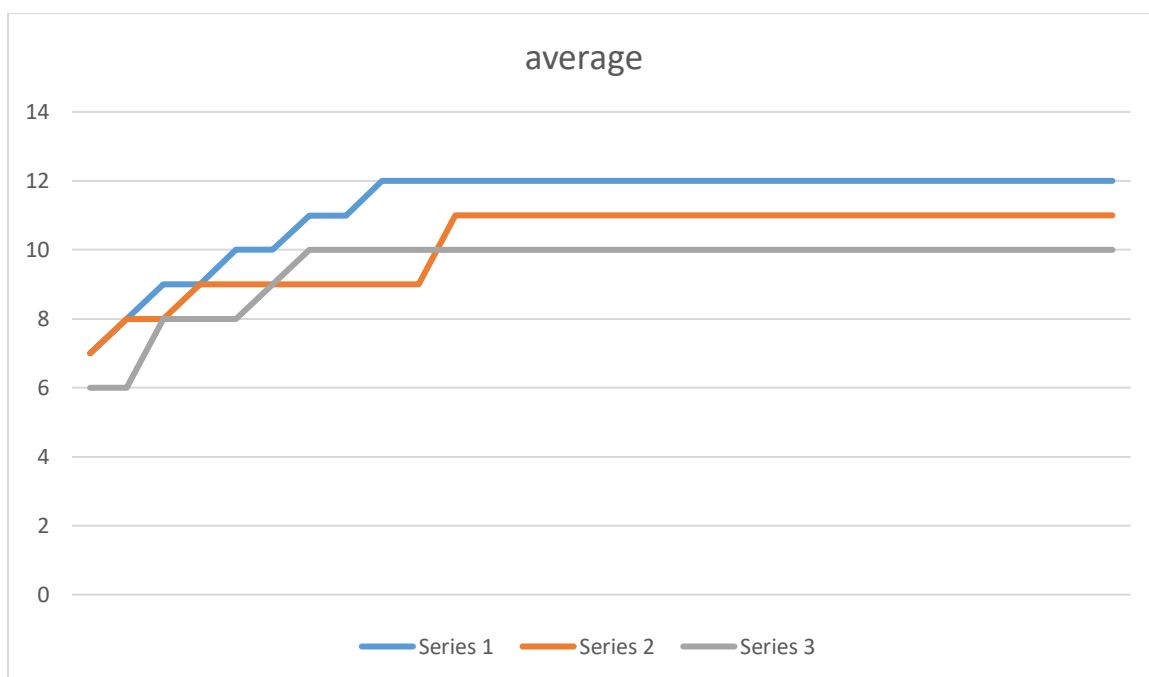
در این حالت 3 احتمال مختلف بررسی می شود:

حالت اول: $1/\text{objectnumber}$

حالت دوم: $1/(\text{MATH.POW}(\text{OBJECTNUMBER}, 0.0002))$

حالت سوم: $1/(\text{MATH.POW}(\text{OBJECTNUMBER}, 100000))$





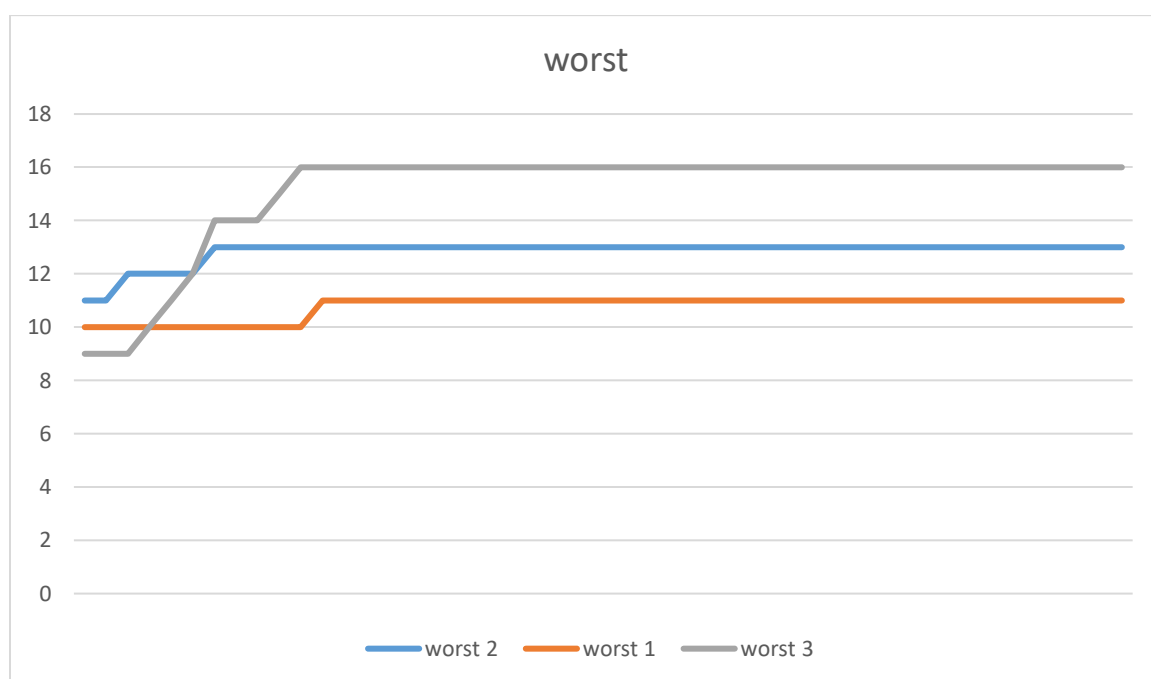
در این جمعیت محدود تغییر احتمال جهش تاثیر به سزایی ندارد ولی در همین حد با افزایش احتمال جهش روند تغییر میانگین و یکسان شدن جامعه بهتر بوده است.

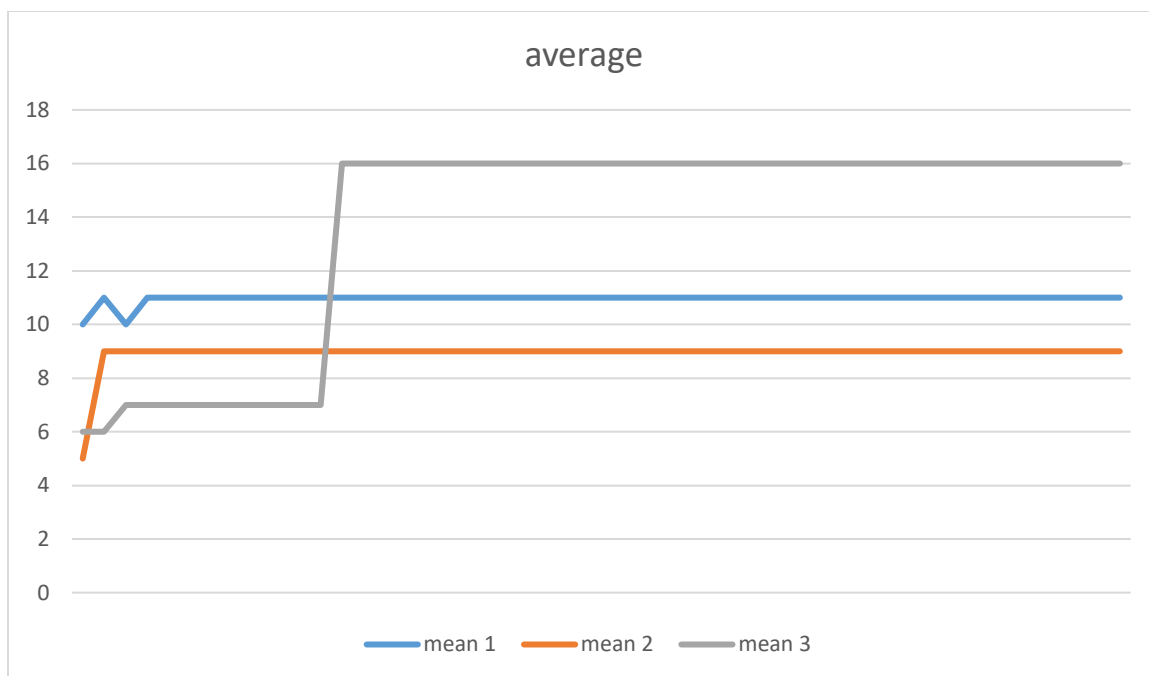
(ج)

نمونه اول N=10

نمونه دوم N=11

نمونه سوم N=12





در مورد بدترین با افزایش همگرایی به حالت نهایی سریعتر رخ داده است. در مورد میانگین حول یک عدد انحراف دارند.

(د)

در این جا سه اندازه جمعیت بررسی شده است.

اندازه=5

اندازه=10

اندازه=30

