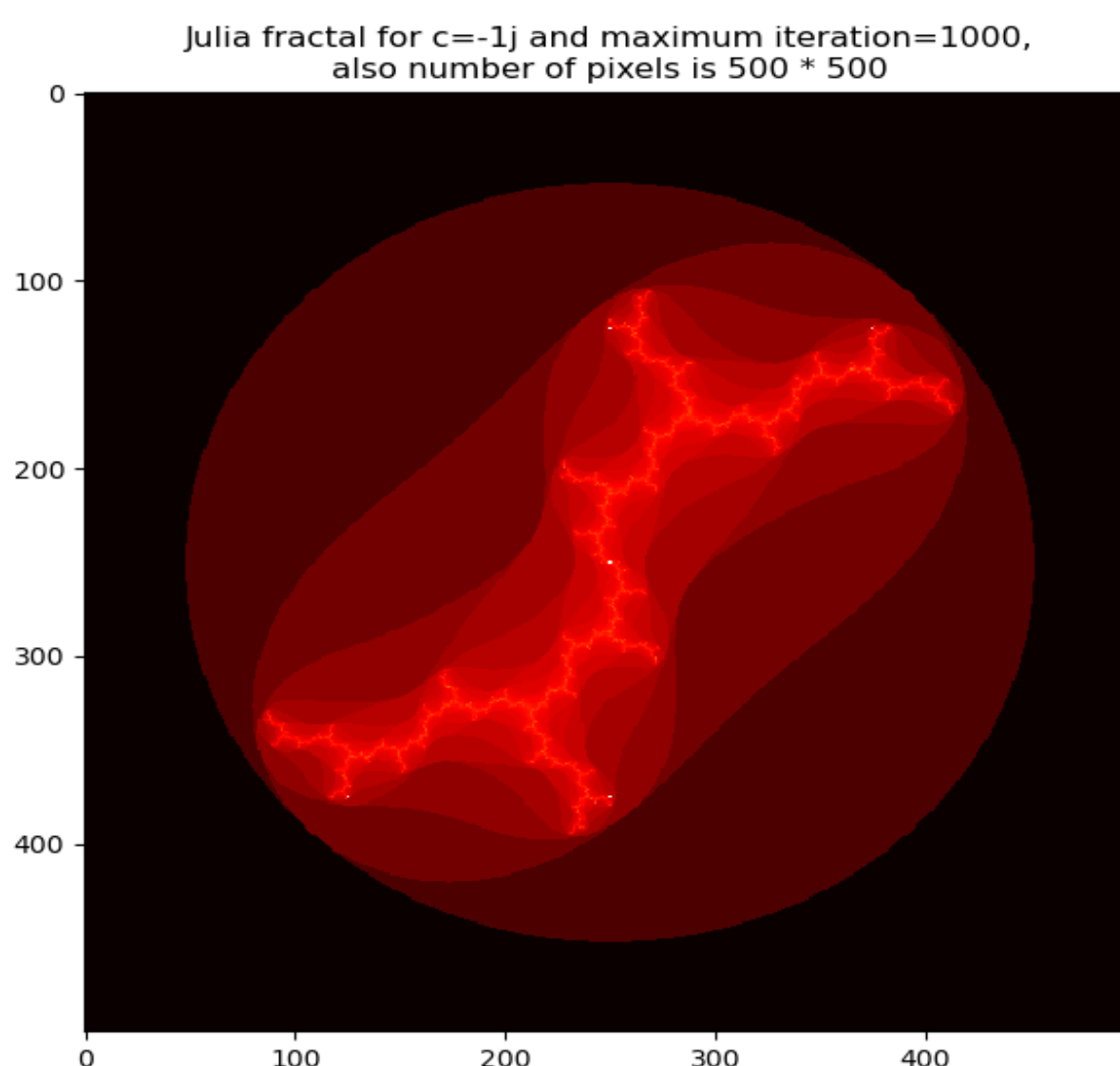
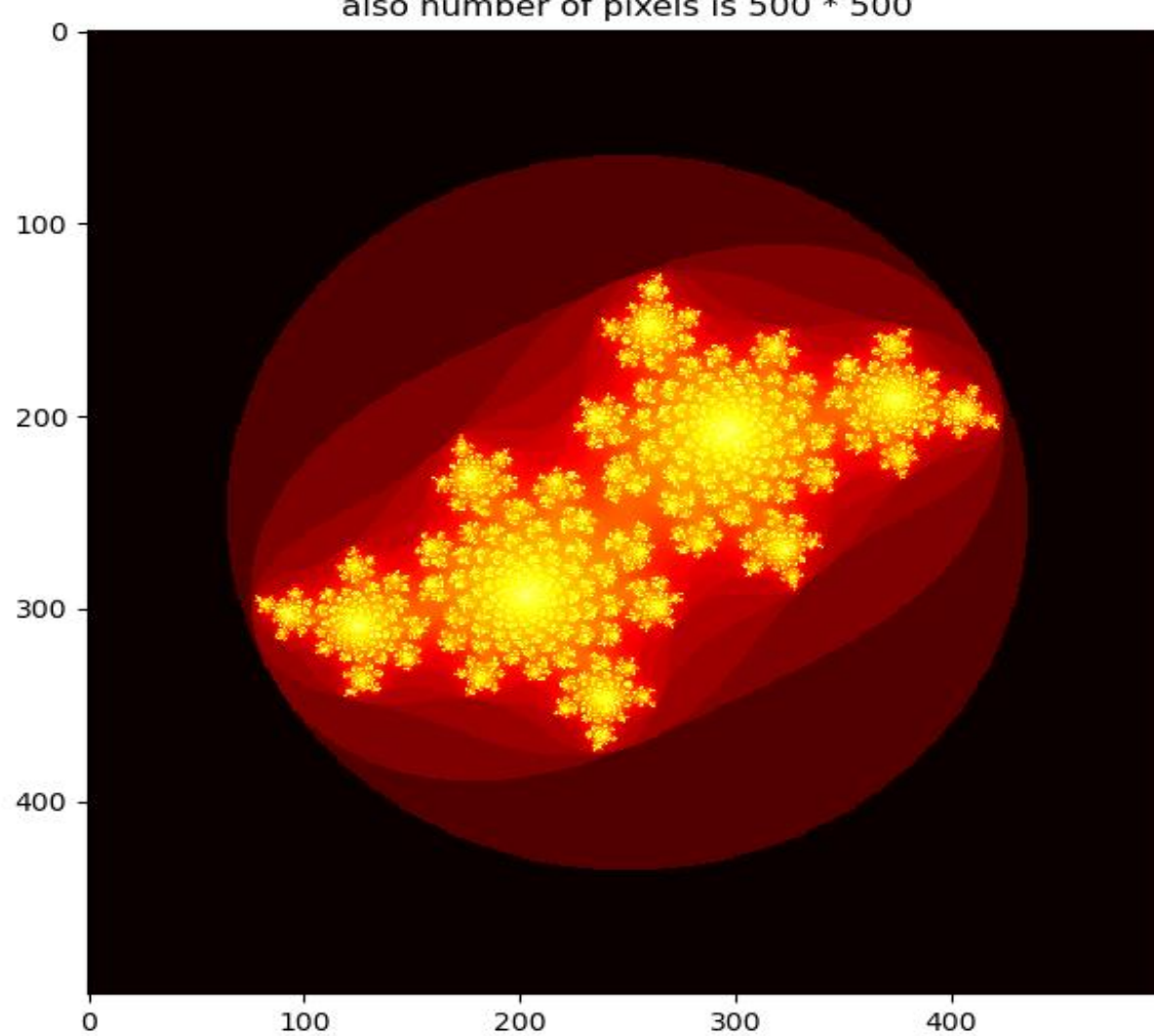


۱. مجموعه‌ی ژولیا:

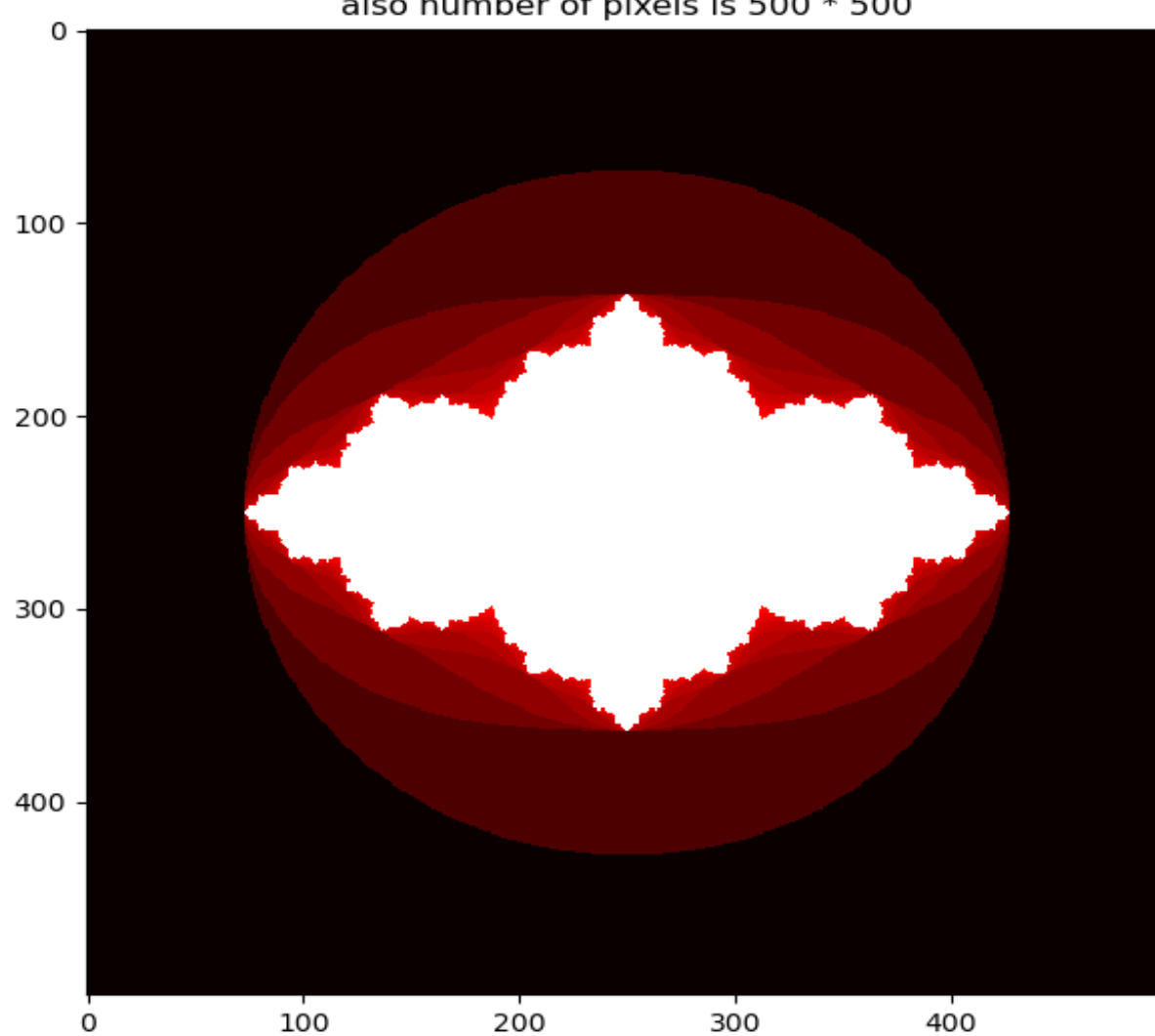
برای تولید مجموعه‌ی ژولیا (فراکتال ژولیا) تابع `showJulia()` را تعریف می‌کنم. ابتدا این تابع را مقداردهی اولیه می‌کنیم. مثلاً تعیین می‌کنیم که صفحه‌ی مختلطی که قرار است بر روی هر نقطه‌ی آن تابع `fz()` را اثر دهیم در چه محدوده‌ی ایکس و وای باید قرار داشته باشد. همچنین مشخص می‌کنیم که در هر راستا تعداد پیکسل‌ها یا در واقع تعداد همان نقاط مختلط صفحه چقدر است. کران بالای تعداد تکرارها را هم مشخص می‌کنیم. فکر می‌کنم از نام متغیرهایی که برای این مقداردهی‌ها استفاده کرده‌ام، کاملاً مشخص است که هر کدام از این پارامترها در دل کدام متغیر قرار گرفته است. در ادامه یک آرایه‌ی دو بعدی که هر بعد آن برابر با تعداد پیکسل‌های صفحه است می‌سازیم و در هر خانه‌ی این آرایه مختصات آن نقطه از صفحه را قرار می‌دهیم. در ادامه تابع `fz()` را بر هر نقطه اثر می‌دهیم. برای این منظور یک حلقه‌ی `while` تعریف شده است. در شرط‌های این حلقه، علاوه بر در نظر گرفتن شرط همگرایی، یک کران‌بالا هم برای شماره‌ی تکرار در نظر می‌گیریم. فایده‌ی این کار این است که از افتادن در دام حلقه‌ی بی‌پایان فرار کنیم. همچنین برای این‌که شکل خروجی مناسبی بدست بیاریم، کران بالای ۱۰۰۰ کفایت می‌کند. در نهایت پس از آن‌که اثردهی تابع بر نقاط صفحه انجام شد، به کمک دستور `imshow()` و استفاده از رنگ‌نگاست `cm.hot` شکل‌های زیبای ژولیا را می‌سازیم و بر روی صفحه‌ی نمایش‌گر نمایش می‌دهیم. یک نکته‌ای که وجود دارد این است که تولید کردن همه‌ی این شکل‌ها زمان اجرای یکسانی ندارند. دلیلش هم احتمالاً به تعداد نقاط همگرایی که برای مقادیر گوناگون پارامتر `C` وجود دارد برمی‌گردد؛ برای برخی شکل‌ها، همگرایی دیرتر رخ می‌دهد و در نتیجه محاسبات بیشتری نیاز است انجام شود. در ادامه برای مقادیر `C` که در جزوه آمده است، مجموعه‌های ژولیای متناظر را می‌سازم و عکس‌های آن‌ها را در ادامه می‌آورم.



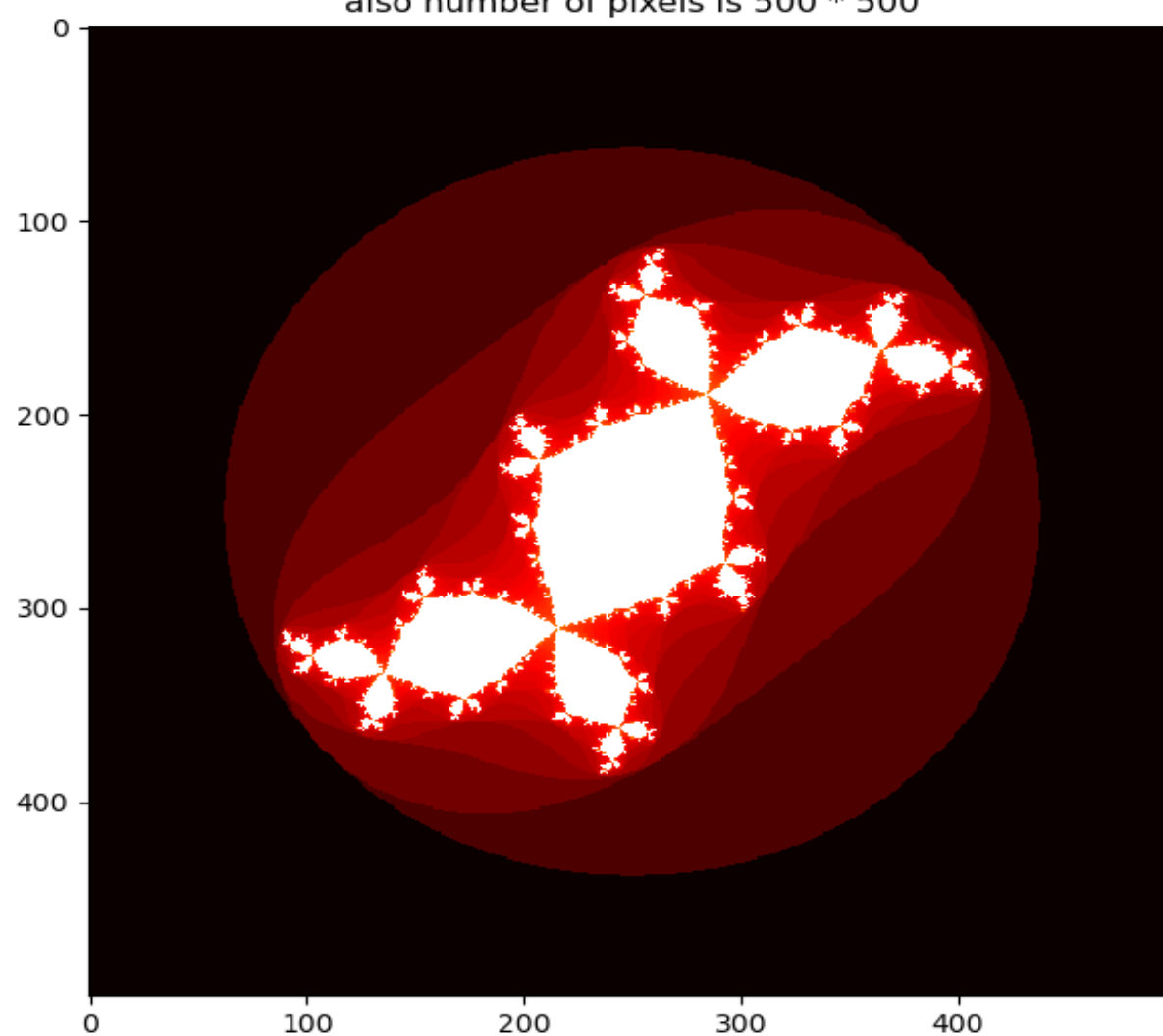
Julia fractal for $c=(-0.4-0.6j)$ and maximum iteration=1000,
also number of pixels is $500 * 500$



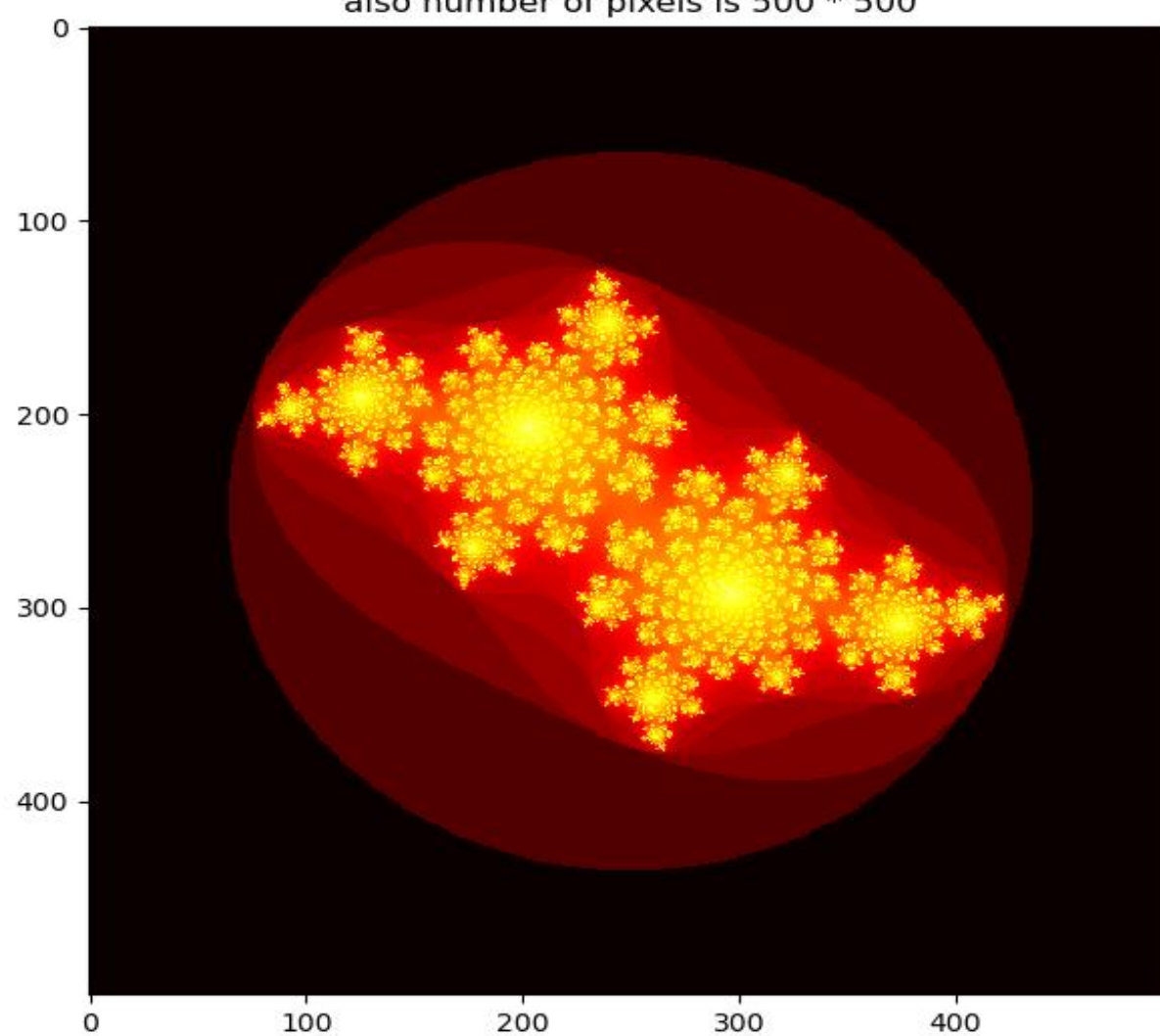
Julia fractal for $c=(-0.6+0j)$ and maximum iteration=1000,
also number of pixels is $500 * 500$



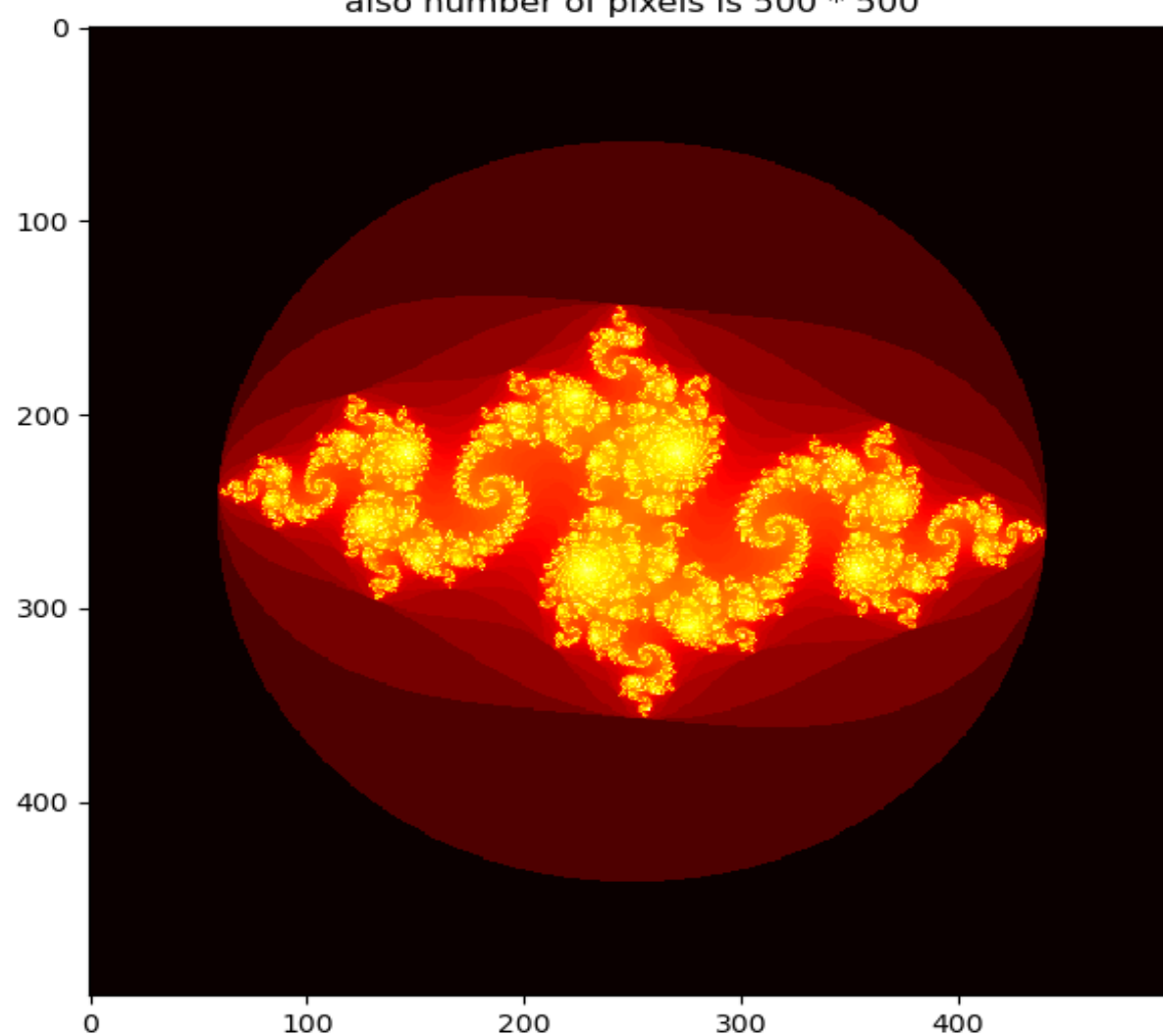
Julia fractal for $c=(-0.12-0.75j)$ and maximum iteration=1000,
also number of pixels is 500 * 500



Julia fractal for $c=(-0.4+0.6j)$ and maximum iteration=1000,
also number of pixels is 500 * 500

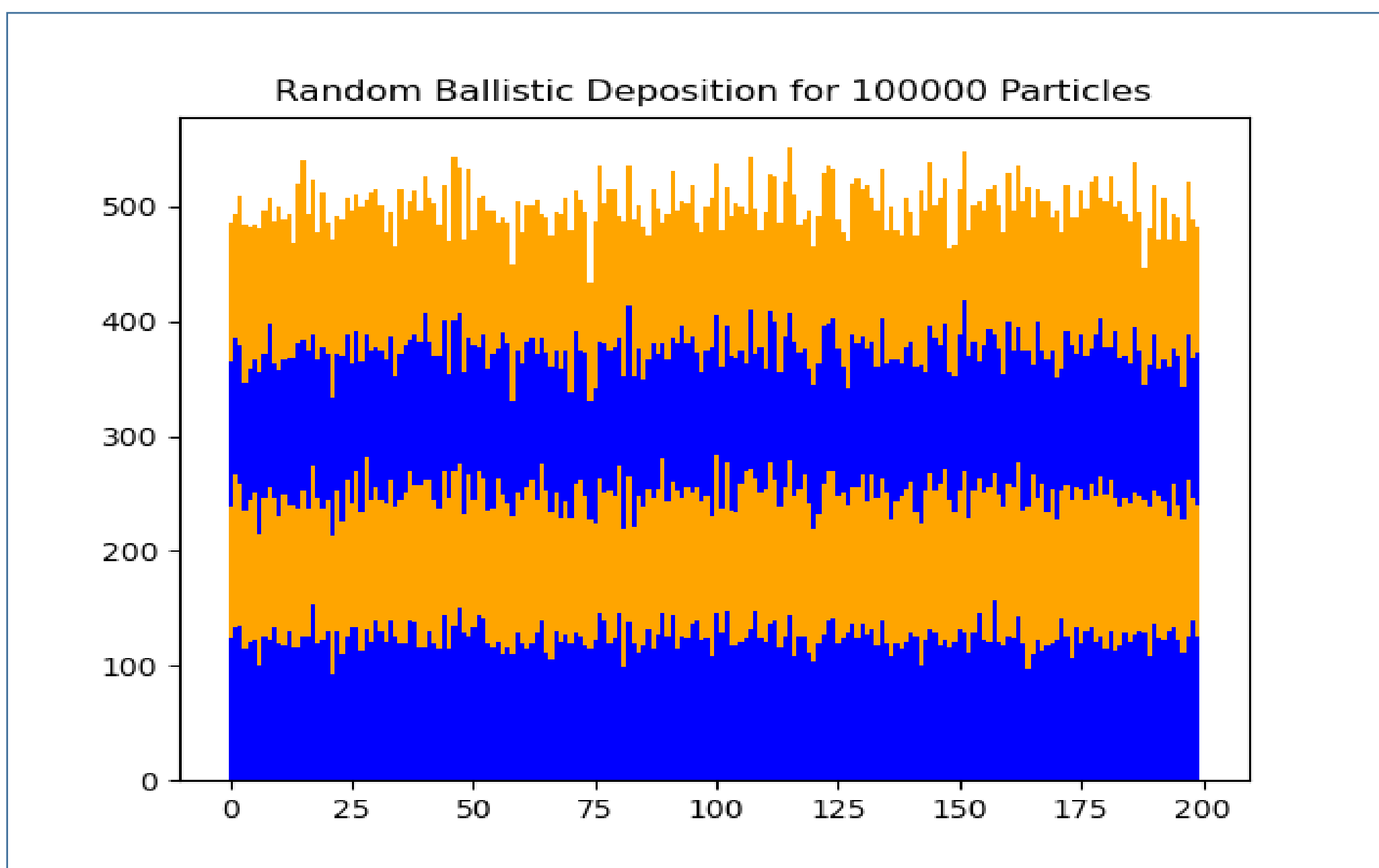


Julia fractal for $c=(-0.8+0.16j)$ and maximum iteration=1000,
also number of pixels is 500 * 500



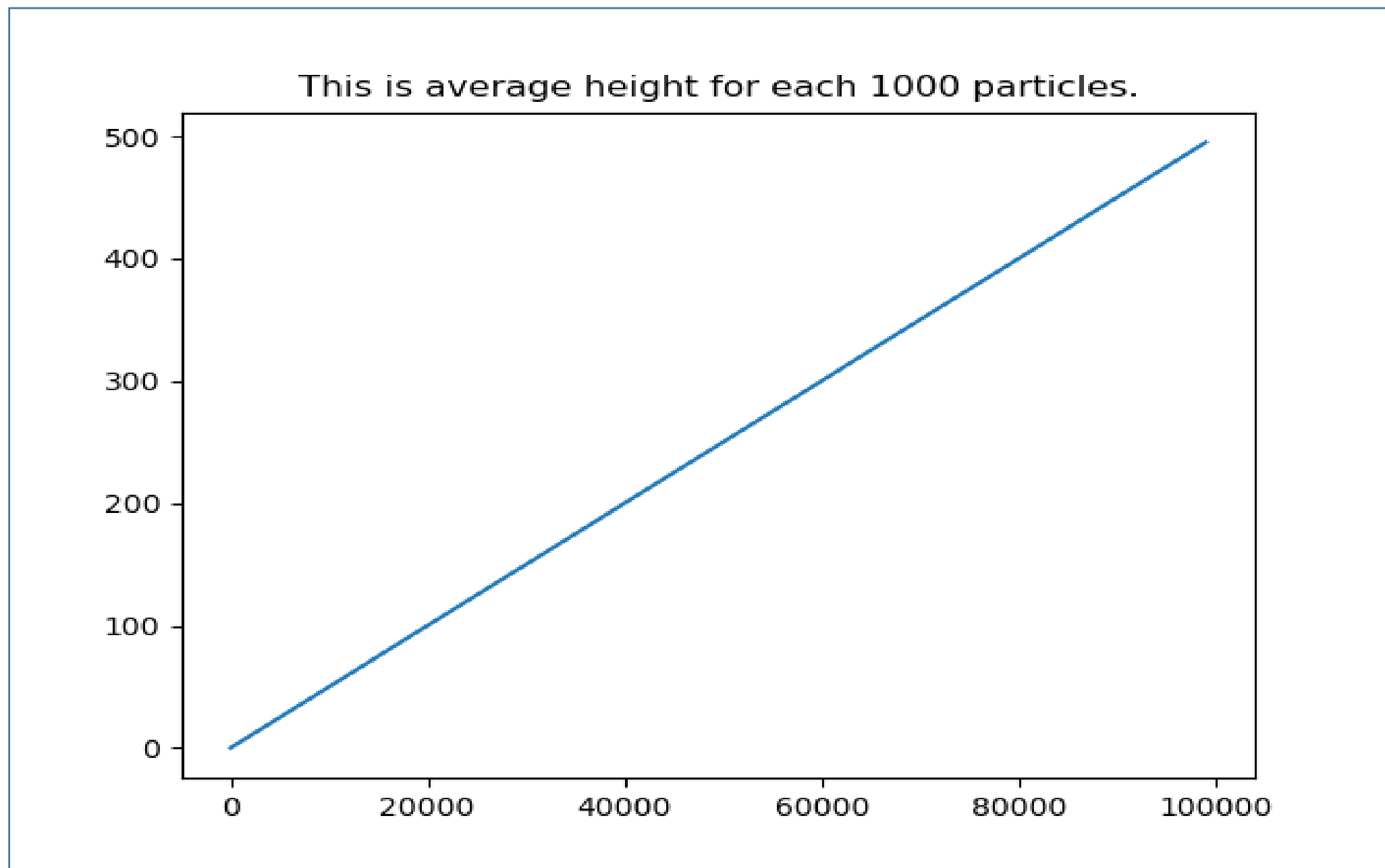
۲. ولنشست تصادفی

شبیه‌سازی دینامیک ولنشست تصادفی به کمک دو تابع `makeLayers()` و `dataCalculations()` انجام می‌شود. تابع نخست نمودار مورد نظر را رسم می‌کند. برای این منظور نیازمند مقداردهی اولیه هستیم. البته در مقداردهی اولیه مقادیر پیش‌فرض در نظر گرفته شده است؛ هرچند کاربر می‌تواند به سادگی مقدارهای آن‌ها را تغییر دهد. در رسم نمودار، ۲ رنگ آبی و نارنجی و ۴ زیرلایه در نظر گرفته‌ام. همچنین این تابع علاوه بر رسم نمودار، اطلاعات لازم را هم جمع‌آوری و در حافظه‌ی رم نگه می‌دارد. این اطلاعات شامل ارتفاع لایه در هر مکان است. در ضمن به نظرم به طور خاص برای این مسئله نیازی به حافظه‌ی سخت نبود و استفاده از حافظه‌ی رم برای منظور ما کفایت می‌کرد. این تابع را به گونه‌ای برنامه‌نویسی کرده‌ام که از آن هم برای محاسبه‌ی ارتفاع برحسب مکان و هم مجذور ارتفاع برحسب مکان بتوان استفاده کرد. به کمک تابع اول شکل زیر را بدست آورده‌ام. در رسم این نمودار مدت زمان ما (یا همان تعداد ذرات) ۱۰ به توان ۵ است.



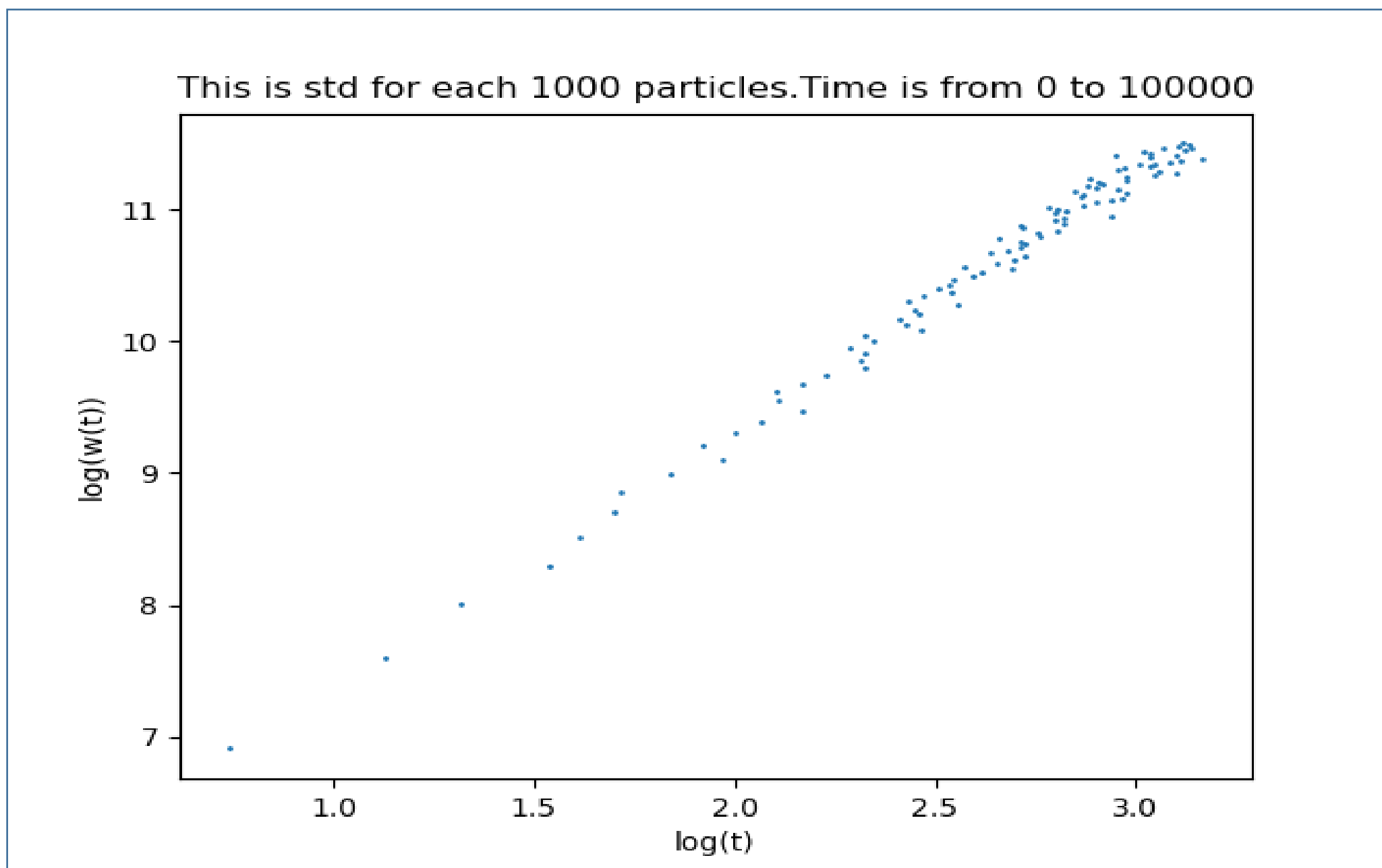
در ادامه تابع `dataCalculations()` با استفاده از خروجی تابع اول، نمودار $\log(w(t))$ برحسب $\log(t)$ و یا نمودار میانگین ارتفاع بر حسب زمان را رسم می‌کند.

در مورد نمودار میانگین بر حسب زمان خروجی به صورت زیر است.



با توجه به اعداد مندرج در نمودار، روشن است که شیب خط برابر با ۰,۰۰۵ یا همان ۱ به روی ۲۰۰ است. با توجه به توضیحات جزوه این موضوع کاملاً بدیهی و البته قابل انتظار است. انجام این کار بدیهی کمک می‌کند تا مطمئن شویم که کدمان به صورت درست کار می‌کند.

در ادامه نمودار $\log(w(t))$ برحسب $\log(t)$ رسم شده است (صفحه‌ی بعد).



متاسفانه همان طور که می بینید برای تعداد ذرات زیاد (یا همان زمان های بزرگ) انحراف نمودار از خط راست مشاهده می شود. متاسفانه این هفته به خاطر مشکلاتی که برایم پیش آمد نتوانستم وقت کافی برای این سری تمرین بگذارم و به همین دلیل نرسیدم مشکل یادشده را برطرف کنم. اما می توان به طور چشمی هم تشخیص داد که مقدار بتا که همان نسبت $\log(w(t))$ به $\log(t)$ است حدوداً برابر با ۰,۵ است. چون محدوده ی تغییرات نمودار افقی ۲ واحد و نمودار عمودی ۴ واحد است که نسبت آن ها برابر با ۰,۵ می شود. البته قاعدتاً باید از ابزارهای curve fitting برای محاسبه ی دقیق تر شیب خط استفاده کنیم.