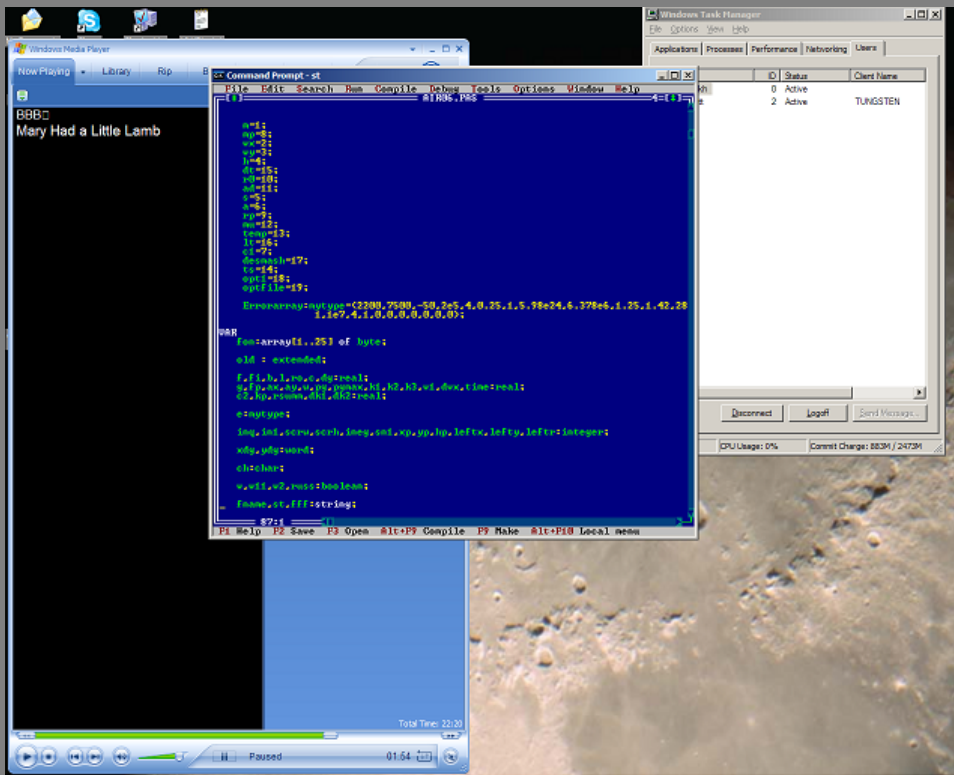


[Get a PDF](#)

Не пугайтесь странному формату этой страницы. Дело в том, что материалы на ней такие старые, что я что-то не смог придумать, как же их более внятно оформить. Речь пойдёт об одной программе, написанной ещё в 1994-м году, и о результатах её работы.

Просто, проводя в очередной раз уборку на своей машине, я раскопал кучу программ, написанных на Паскале начиная с 91-го года, и решил попробовать их запустить. Очень интересно и полезно, кстати, читать свой код многолетней давности и изучать, как изменилось собственное мышление за такой период времени.

Turbo Pascal 7 сегодня -- freeware, так что с загрузкой и установкой его проблем никаких не возникло. Запуск, правда, был не столь гладок: CRT-шный модуль отказывался инициализироваться на слишком быстрых современных машинах, но для этой проблемы нашёлся patch, и вот, о чудо, в 2009-м году на 2003-м Windows Server-е взлетает (в окошке!) Turbo Pascal, компилирует программы, написанные мною ещё в начале 90-х, и исполняет их!



Не правда ли, занятно? Вот Вы читаете эту страницу с моего сервера, а он в это время, быть может, Паскаль крутит? :)

Впрочем, рассказ о другом. О том, как в далёком 94-м году Денни Гурский написал программку для расчёта и

отрисовки множеств Жюлиа. Очень красивые получались картинки. Ну а я занялся тем, что с её помощью исследовал поведение этого множества на ряде других аналитических функций, из чего получались ещё более интересные картинки.

Настоящим исследованием это назвать нельзя (в лучшем случае -- лишь полётом математической интуиции), да и строгости в нём нет никакой. Может быть, мы и могли бы довести его до ума 15 лет назад, но не довели, и всё, что сегодня осталось -- это идея и множество цветных картинок.

О них, собственно, и пойдёт речь.

В программе множество Жюлиа на комплексной плоскости определялось как множество таких точек  $Z$ , в которых последовательность отображений

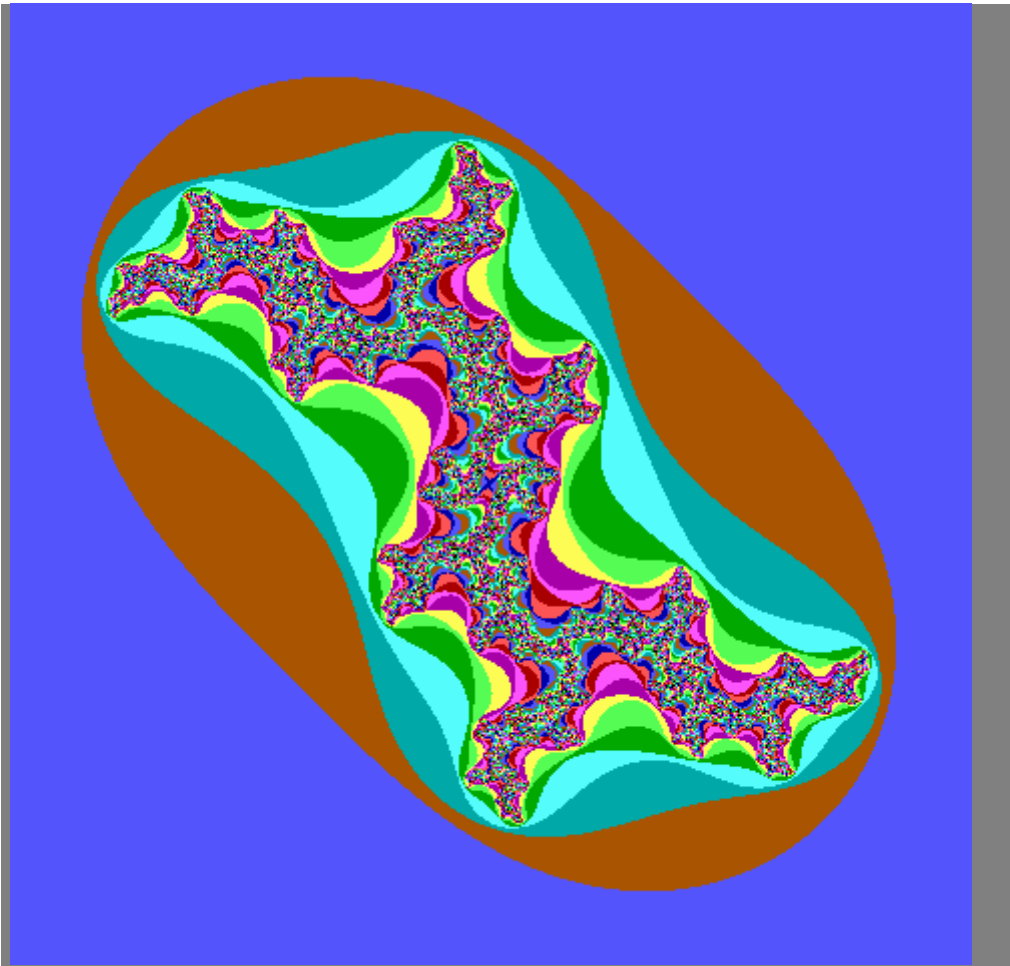
$$Z_{n+1} = Z_n^2 + c$$

не стремится к бесконечности при некотором заданном комплексном  $c$  при  $n$  стремящемся к бесконечности. Это определение, вообще говоря, несколько отличается от классического, но у нас в те времена не было Википедии и Интернета, чтобы свериться, не вылезая из-за компьютера :)

На практике, разумеется, до бесконечности никто не считал. Если последовательность ещё не расходилась при некотором "пороговом"  $n$ , точка окрашивалась в чёрный цвет. Если расходилась -- в другой, в зависимости от номера шага, на котором это произошло. Расходимость определялась как превышение  $|Z|$  некоторой константы.

(Заметьте, что всё это вычислялось на 286-х машинах, так что алгоритм вынужденно был очень простым. Ибо даже картинка 200x200 могла отрисовываться весьма небыстро. Сегодня тот же самый код работает раз в 100 шустрее).

То, что получалось, было изображением даже не самого множества Жюлиа, но точек, к нему прилегающих. Тем не менее, в зависимости от выбора  $c$  результат мог выглядеть достаточно красиво и очень по-разному. Например, вот что выдавалось при  $c = -0.82i$ :

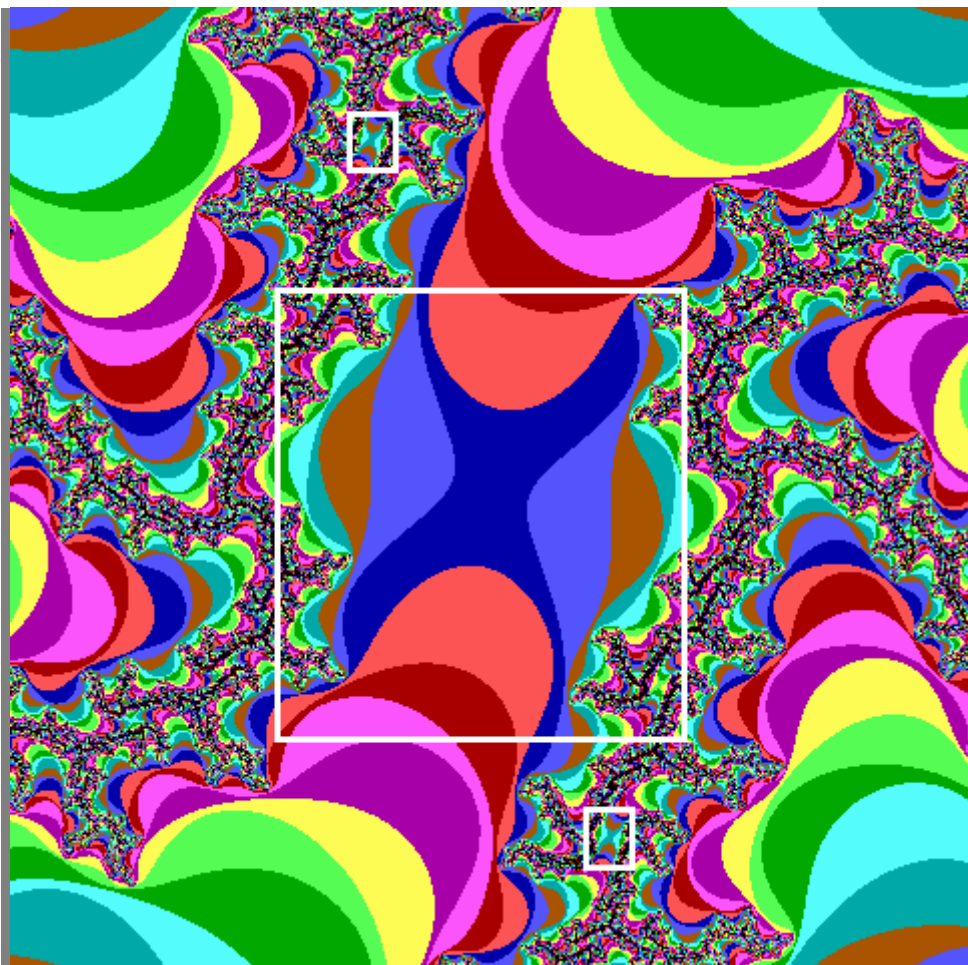


Это множество фрактально. То есть, разглядывая его со сколь угодно большим увеличением, мы будем обнаруживать элементы, во всём подобные по форме более крупным. И так вглубь до бесконечности.

Вот, скажем, центральная часть показанной выше картинка:

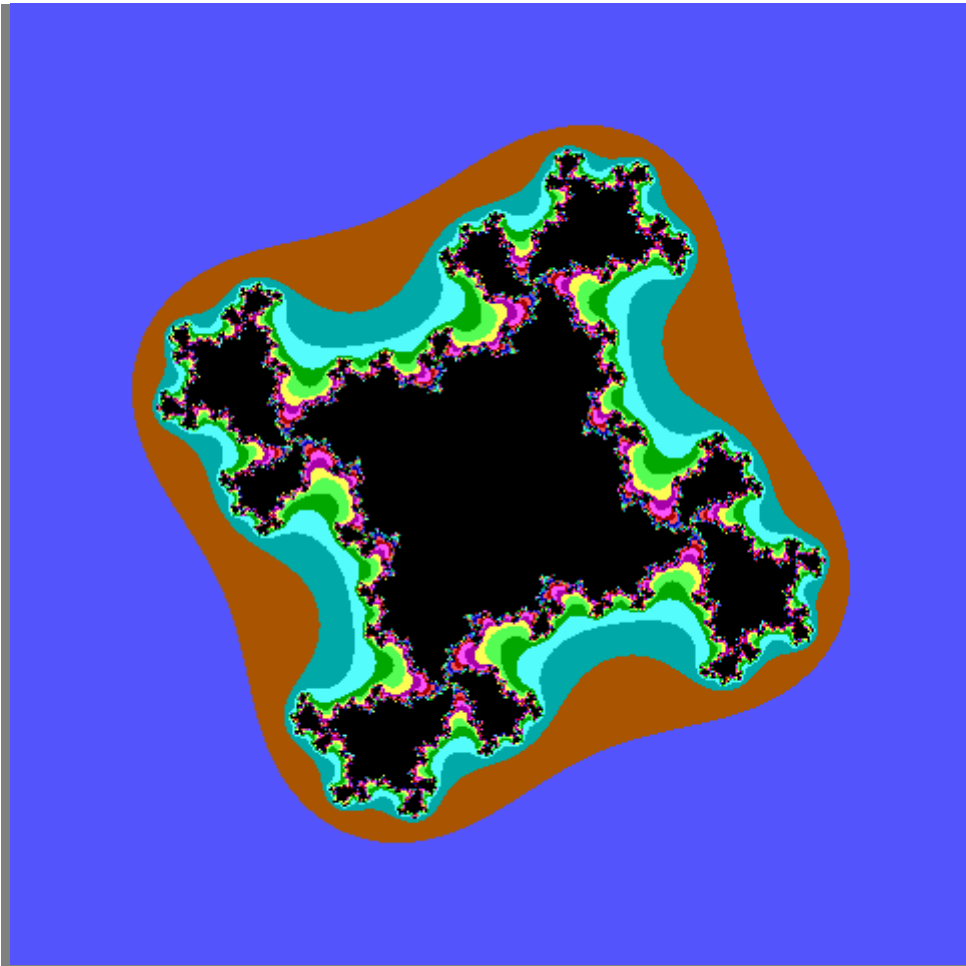


Если увеличить её ещё сильнее, можно разглядеть маленькие "сходящиеся мосты", подобные большому в центре (они выделены белыми квадратами):

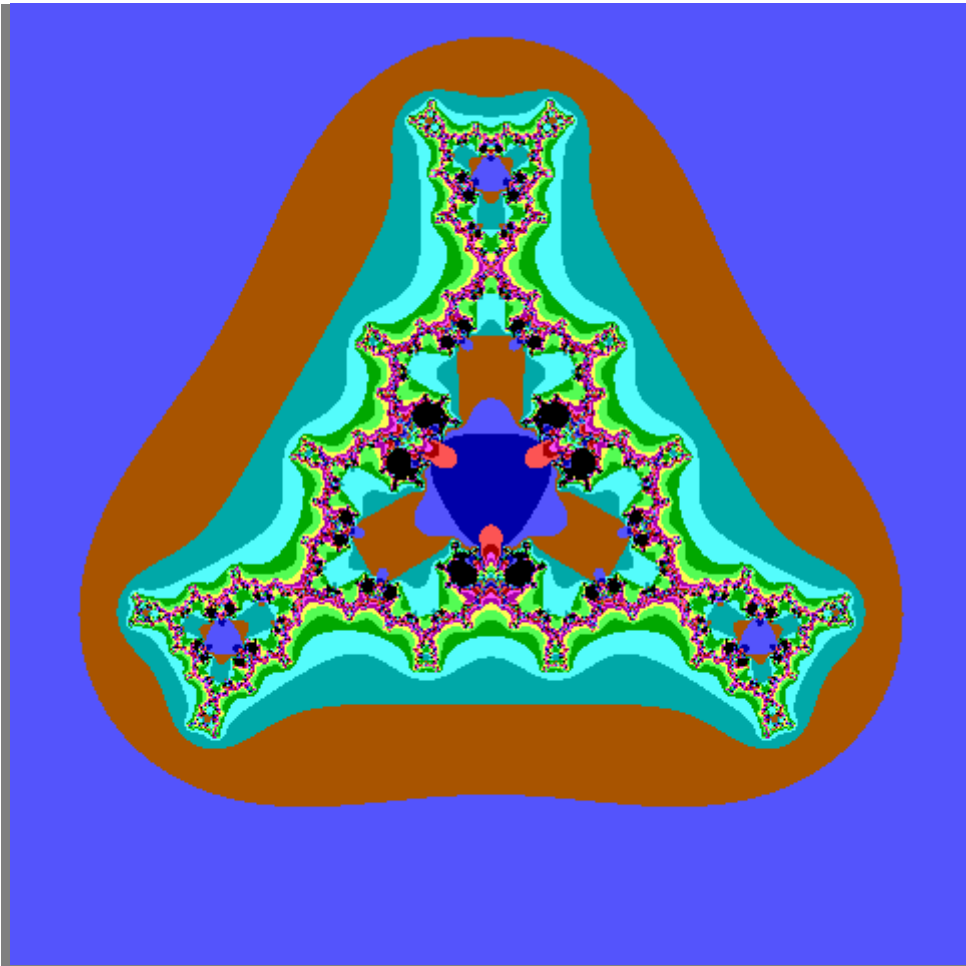


Ещё в 93-м году меня заинтересовал вопрос: а обязательно ли использовать для построения множества Жюлиа исключительно лишь квадратичное отображение? Ведь очевидно же, что этот процесс обобщается как минимум на четвёртую степень (ибо это есть ни что иное, как то же квадратичное отображение, где добавление константы через шаг пропускается). При этом даже критерий расходимости менять не надо, он остаётся строго тем же.

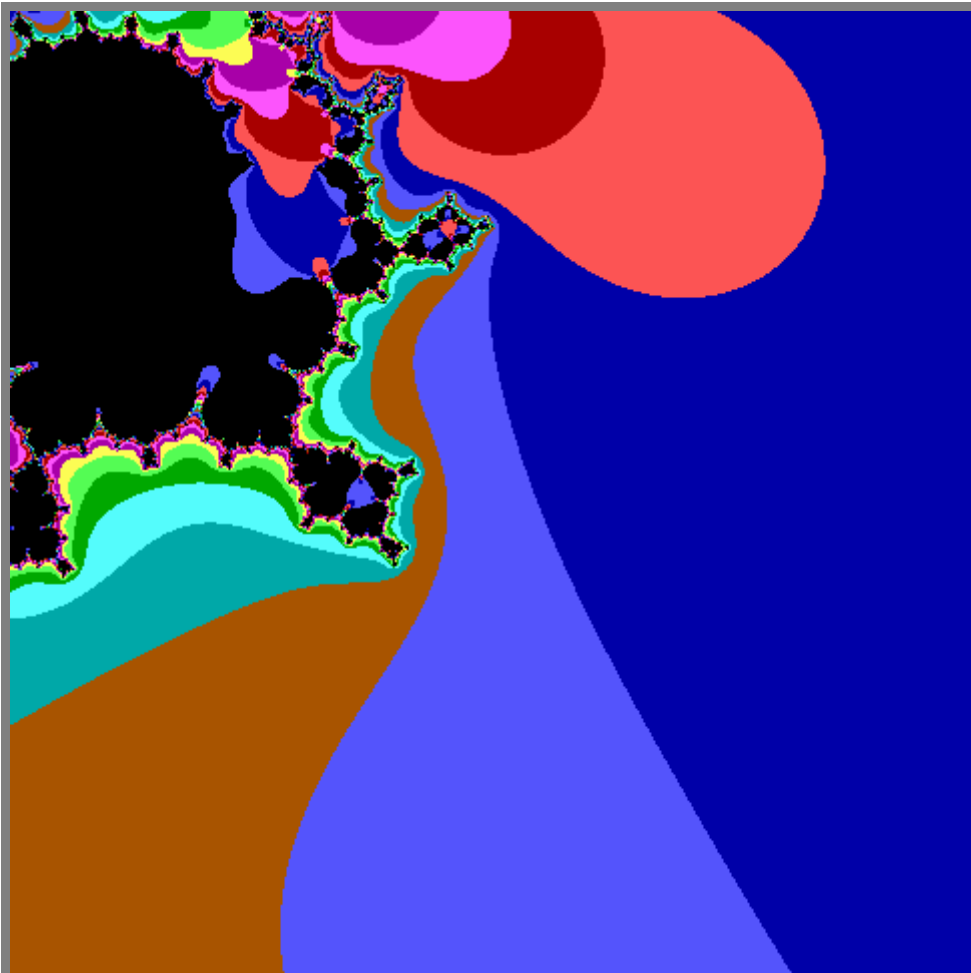
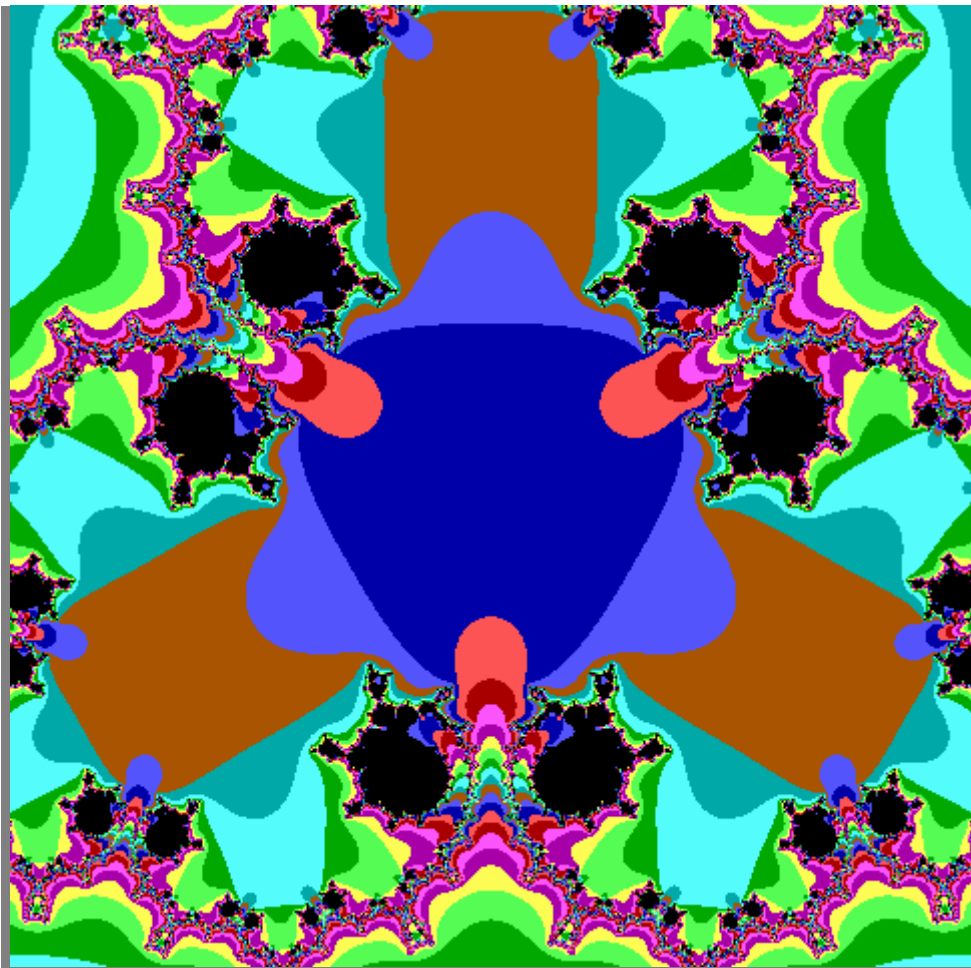
И вот году в 94-м я взял и попробовал. Действительно, для  $z_{n+1} = z_n^4 + c$  при  $c = 0.82i$  получается вот такая картинка:



Следующий логический шаг -- обобщение на произвольную целочисленную степень. Он уже менее очевиден в силу недоказанности применимости критерия расходимости. Но мне было лень проверять на бумаге. На дворе начиналась новая эра "дешёвых" вычислений. Я просто запустил программу, задав кубическое отображение  $z_{n+1} = z_n^3 + c$  -- и вот что вышло ( $c = -1.095i$ ; показана область вокруг центра координат размером 1.6 на 1.6):

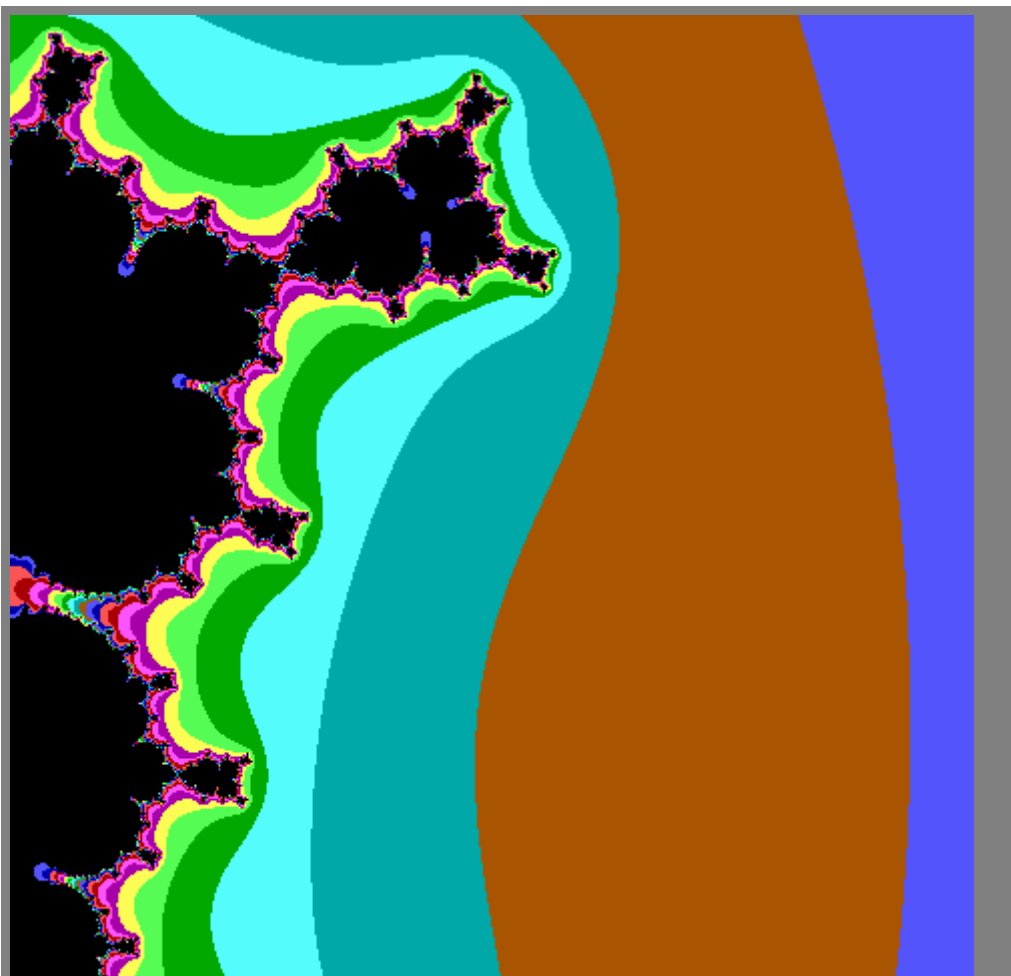


Работает! Причём видна трёхлучевая симметрия. И совершенно честный фрактал, как можно убедиться, постепенно забираясь всё глубже и глубже:



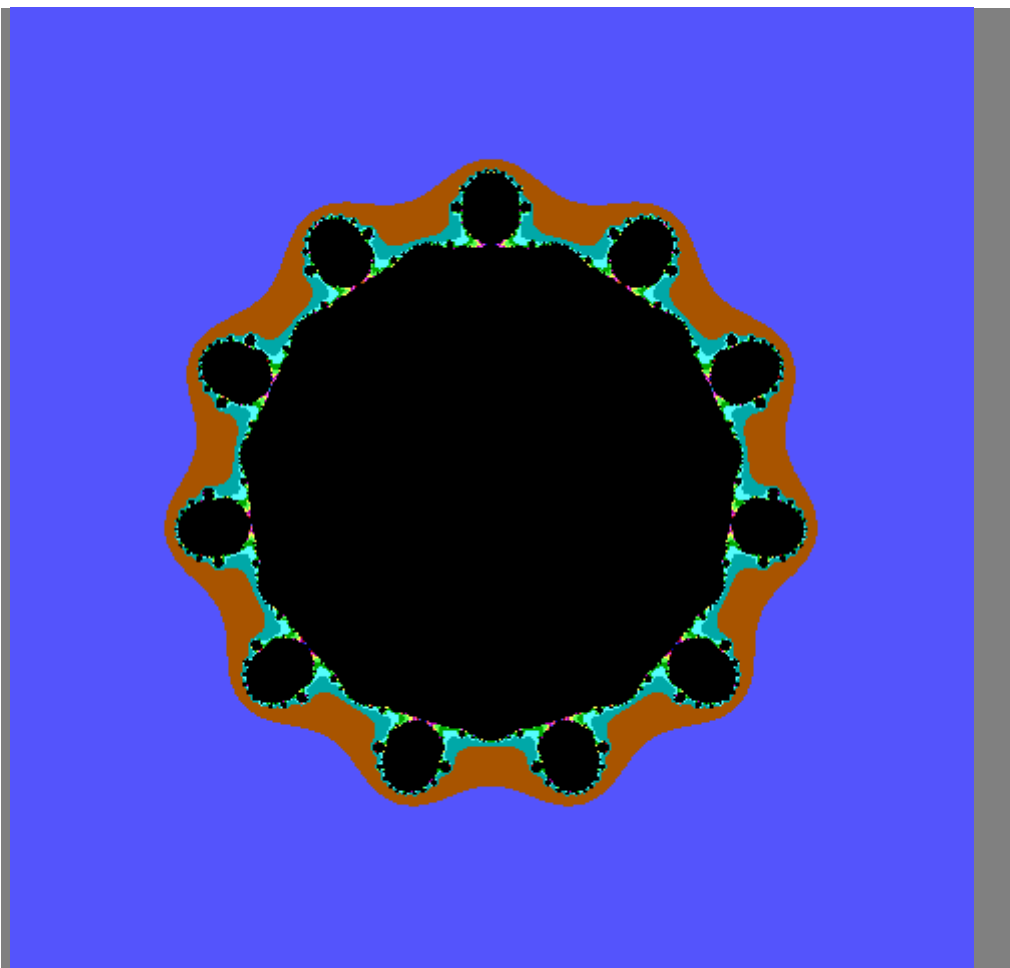






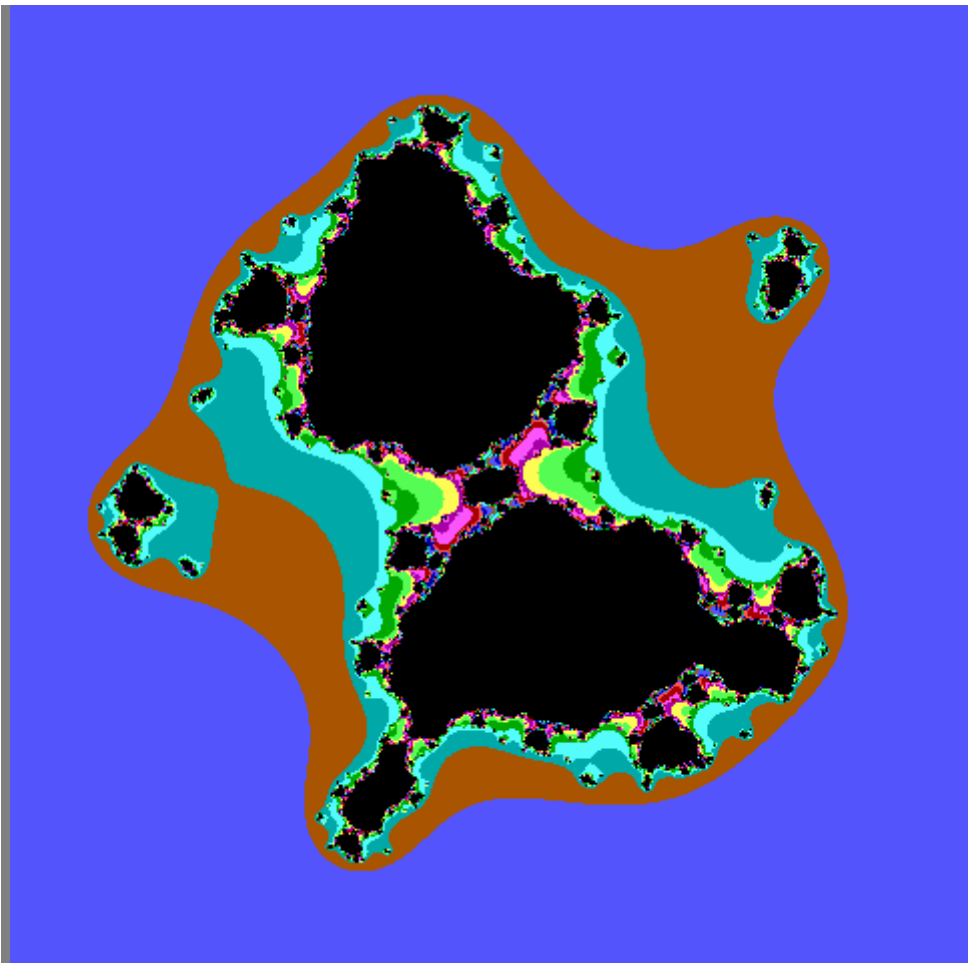
Подмечена закономерность: отображение  $z_{n+1} = z_n^N + c$  порождает  $N$ -лучевую симметрию. Проверим?

Действительно, при  $N = 11$  получается 11 сегментов (квадрат 1.6 на 1.6,  $c = -0.9i$ ):



Хорошо! Похоже, это работает для любой целочисленной степени. А что насчёт суммы двух степеней?

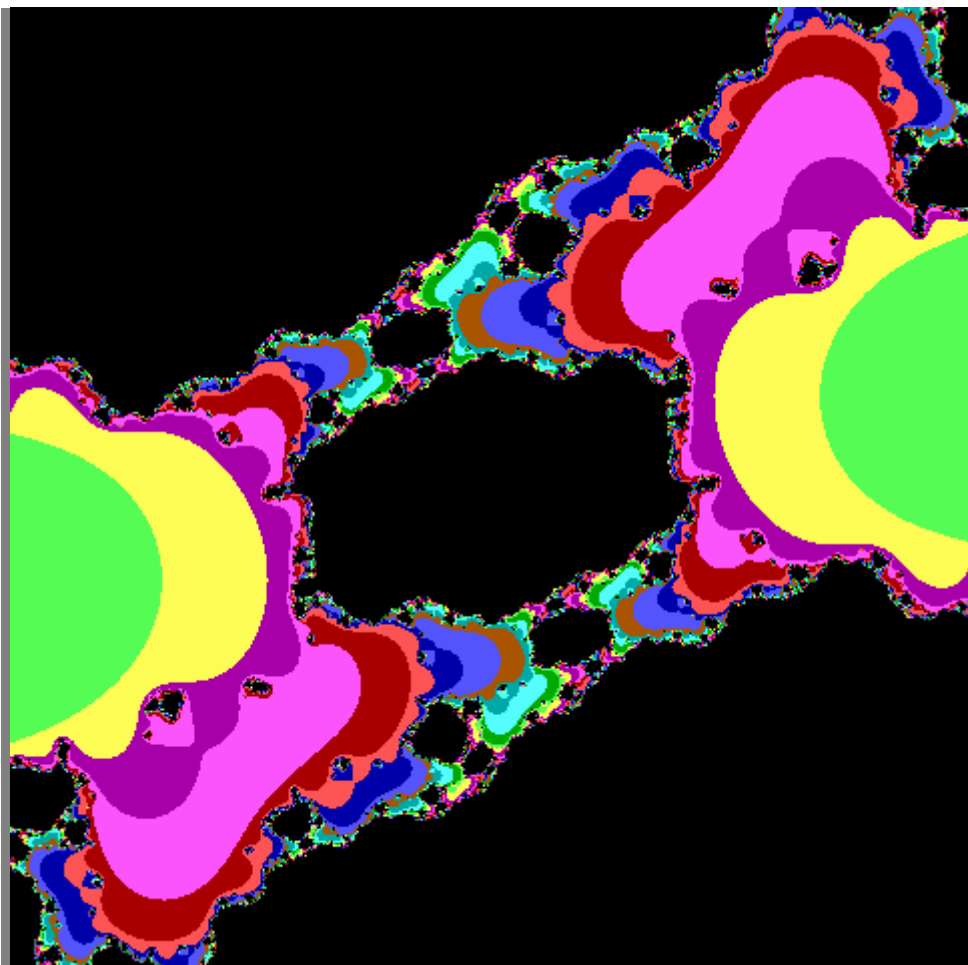
Проверка показывает, что тоже получается:



Здесь использовано отображение  $z_{n+1} = z_n^7 + z_n^2 + c$  при  $c = -0.7i$ .

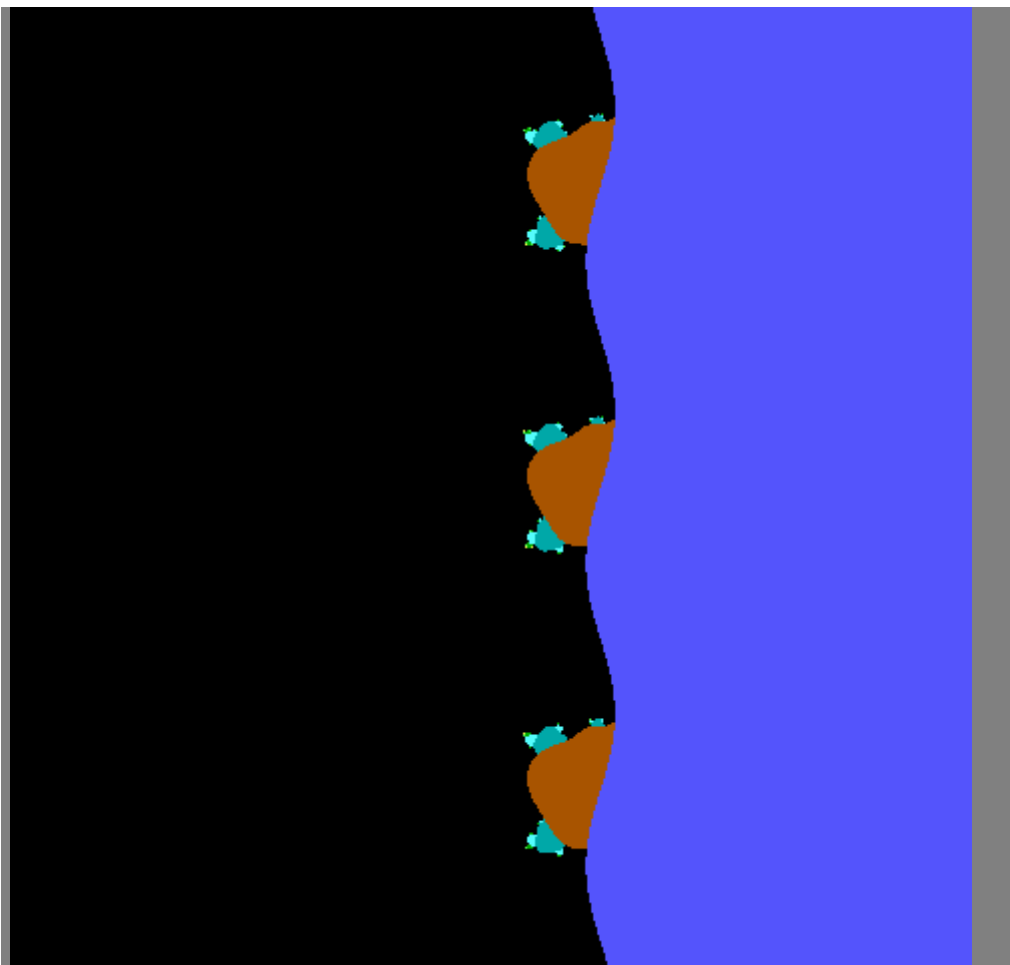
Занятно, что 7-ми и двухлучевая симметрии как будто соревнуются за влияние и в итоге разделяют его сферы: крупным структурам достаётся 180-градусная симметрия, а мелким -- 7-ми-лучевая.

Множество сохраняет фрактальность и разделение симметрий и в малых деталях (область у центра):



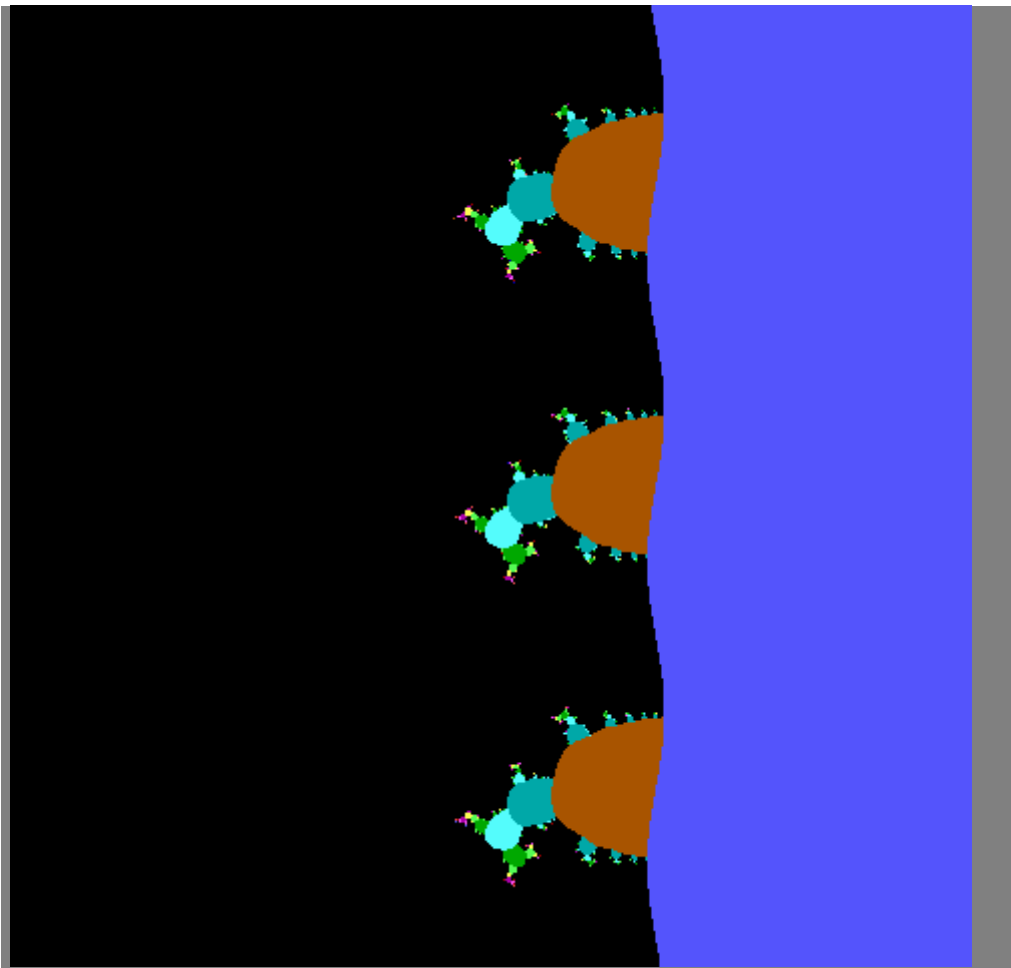
Прекрасно, говорим мы. Раз это работает даже для многочленов, то нельзя ли это обобщить на многочлены бесконечной длины? То есть, на ряды Тэйлора? Пробуем!

$Z_{n+1} = \exp(Z_n) + c$  при  $c = 3.1i$ , квадрат 10 на 10 в центре координат:

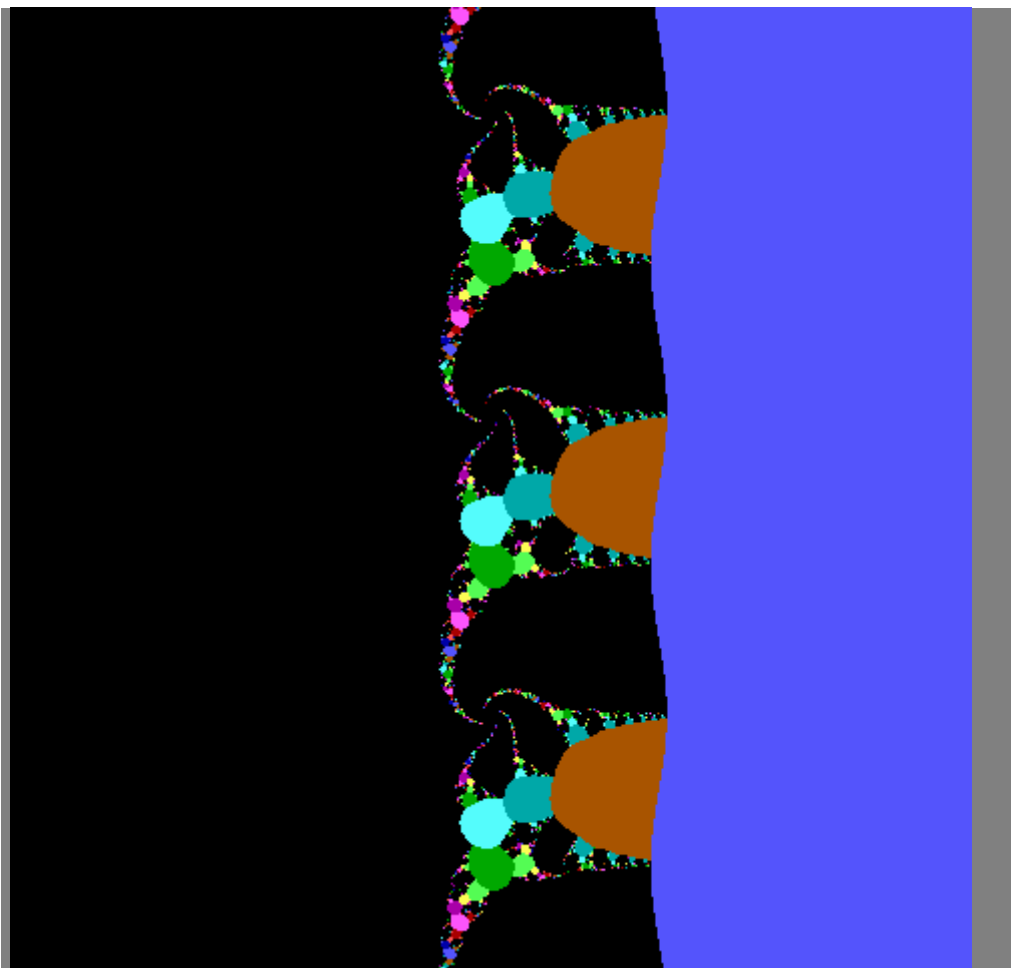


С эстетической позиции не особо впечатляет... но! Есть определённое сходство с картинкой про 11-ю степень. Это же... круг с центром в минус полубесконечности, и радиусом в половину бесконечности! И, на первый взгляд, тоже фрактальный. Ага! Попробуем покрутить константу?

$c = 5.5i$  -- уже лучше:

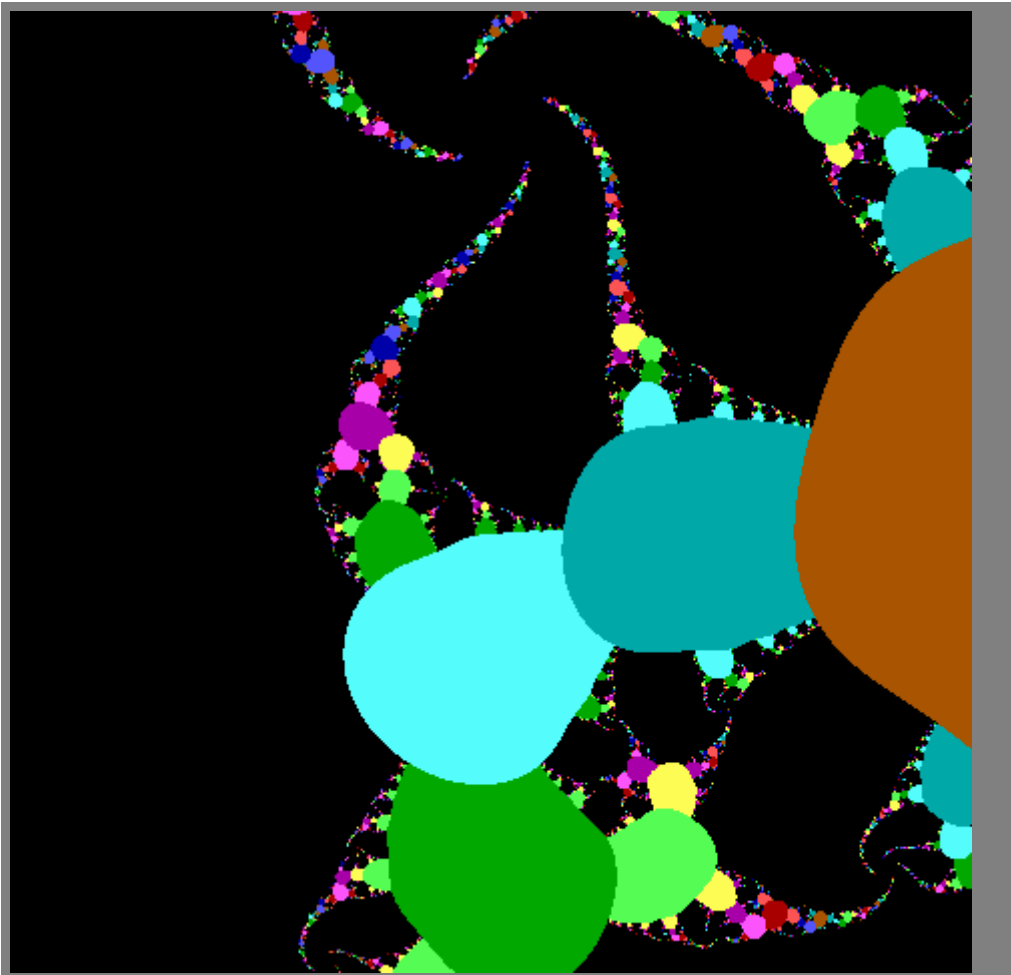
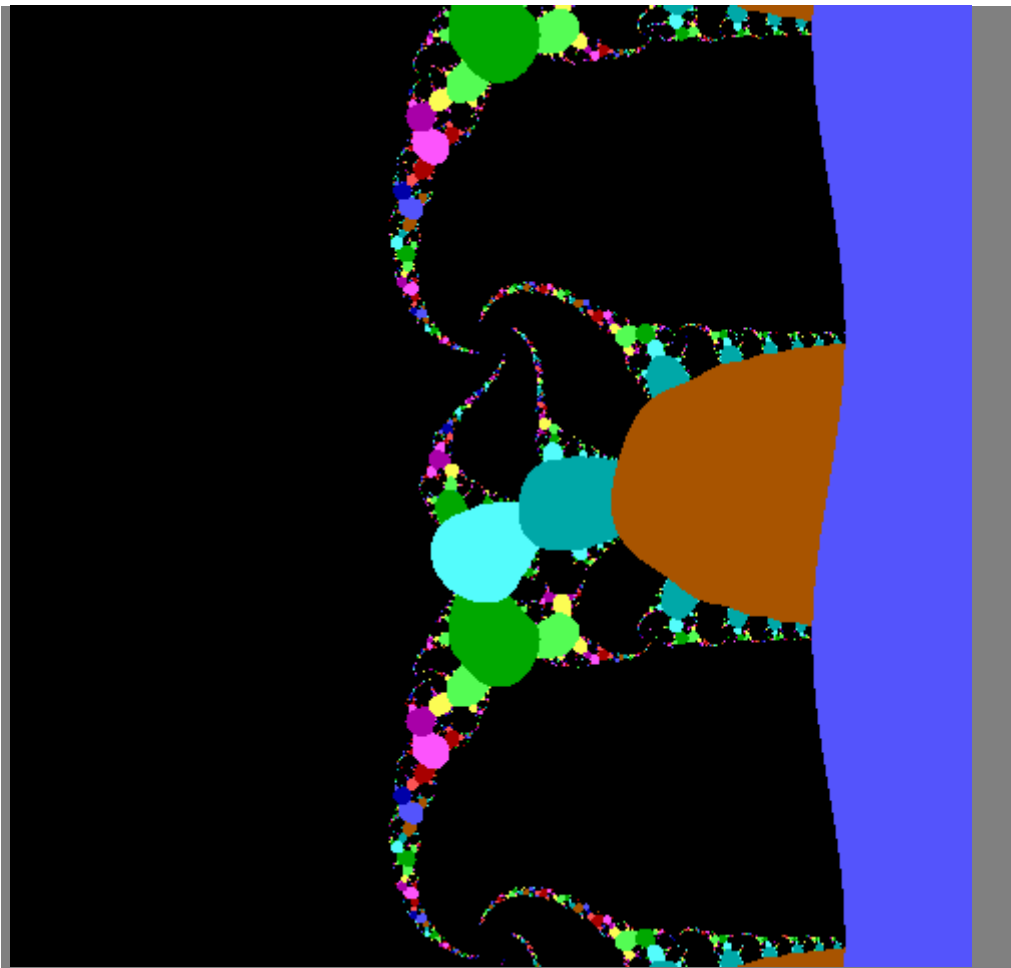


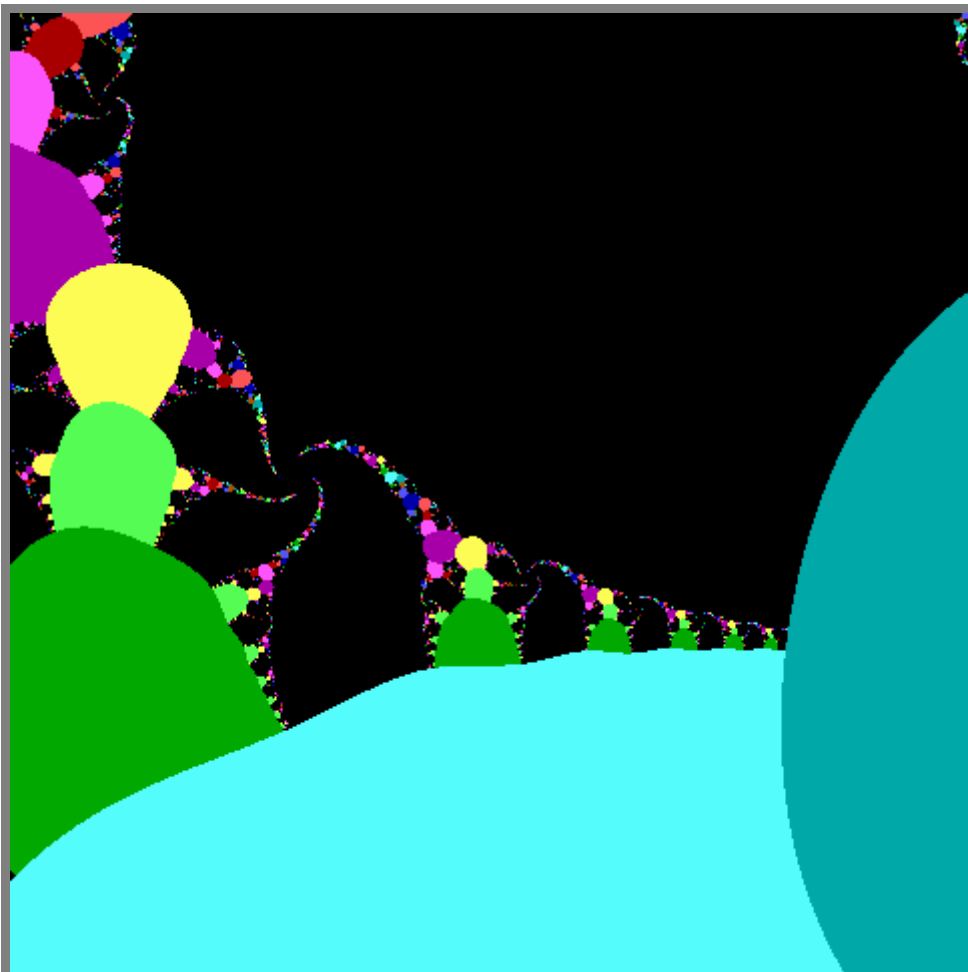
$c = 5.8i:$



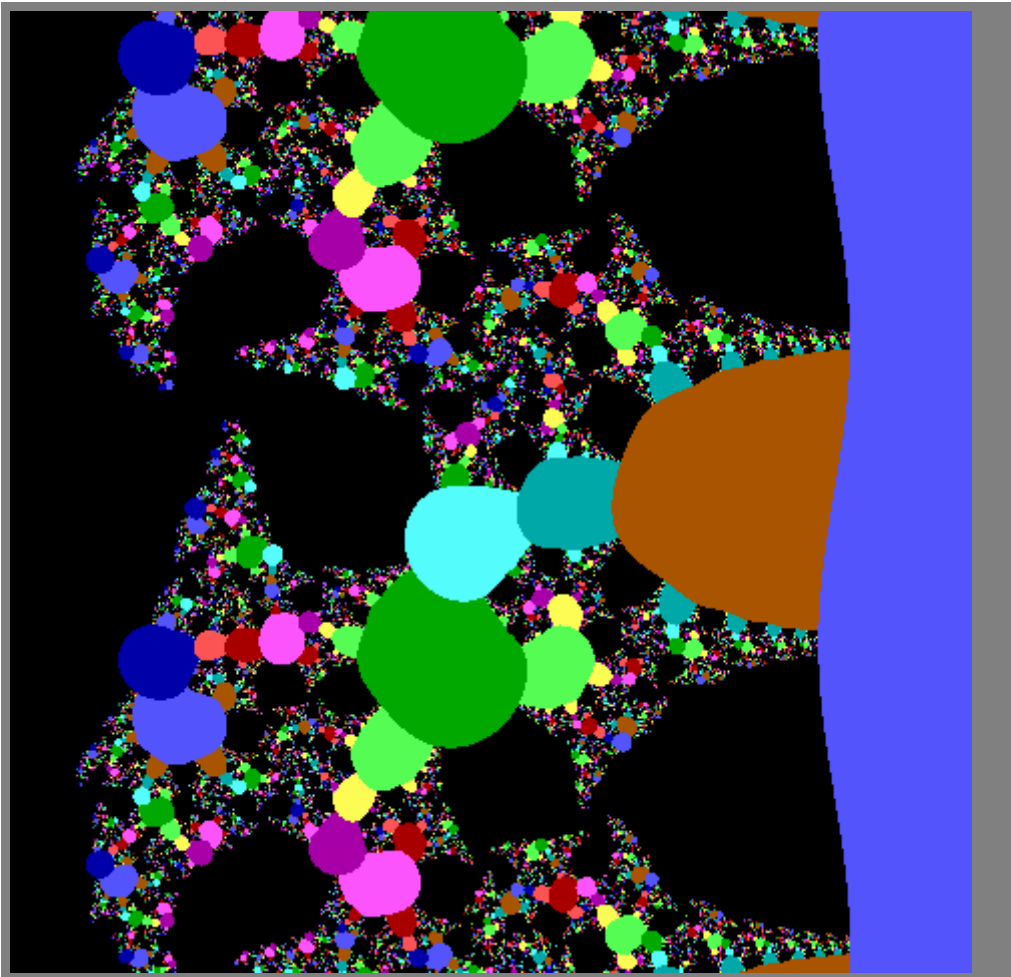
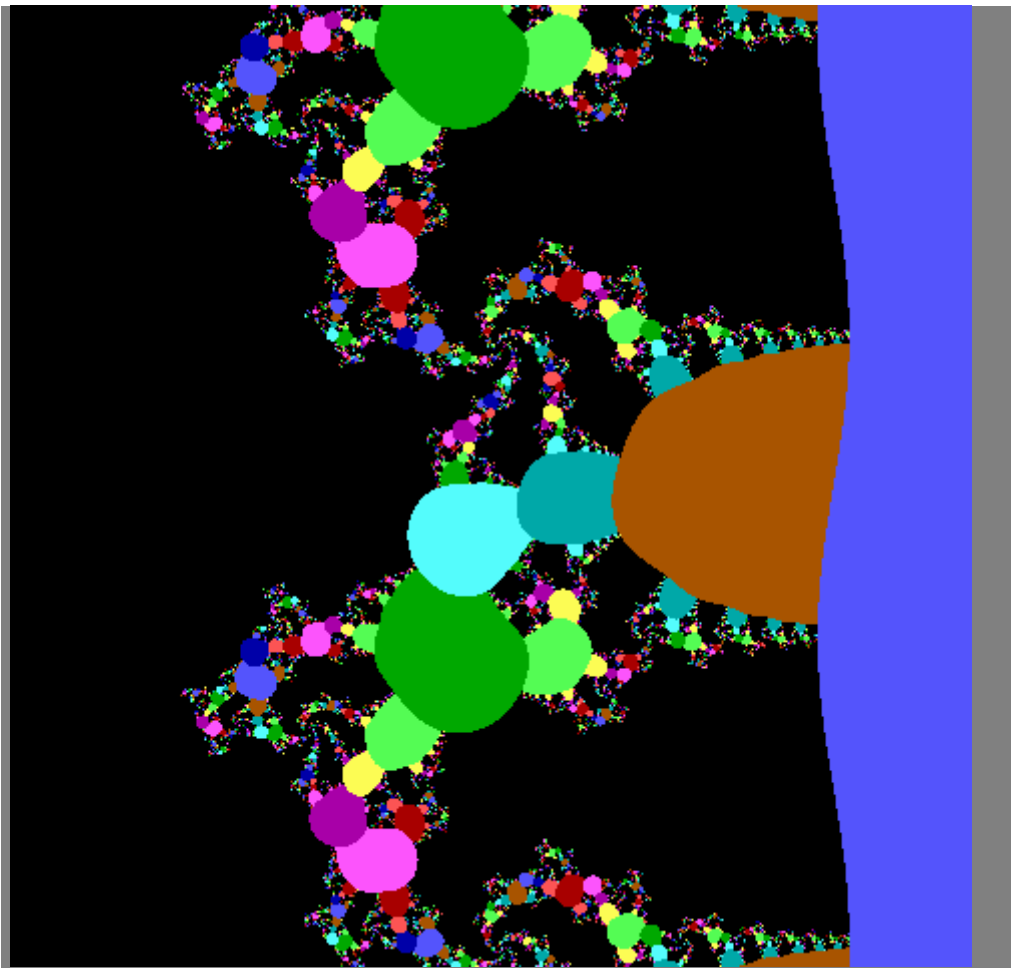
Следующие несколько картинок -- последовательный "наезд" на ту, что выше:

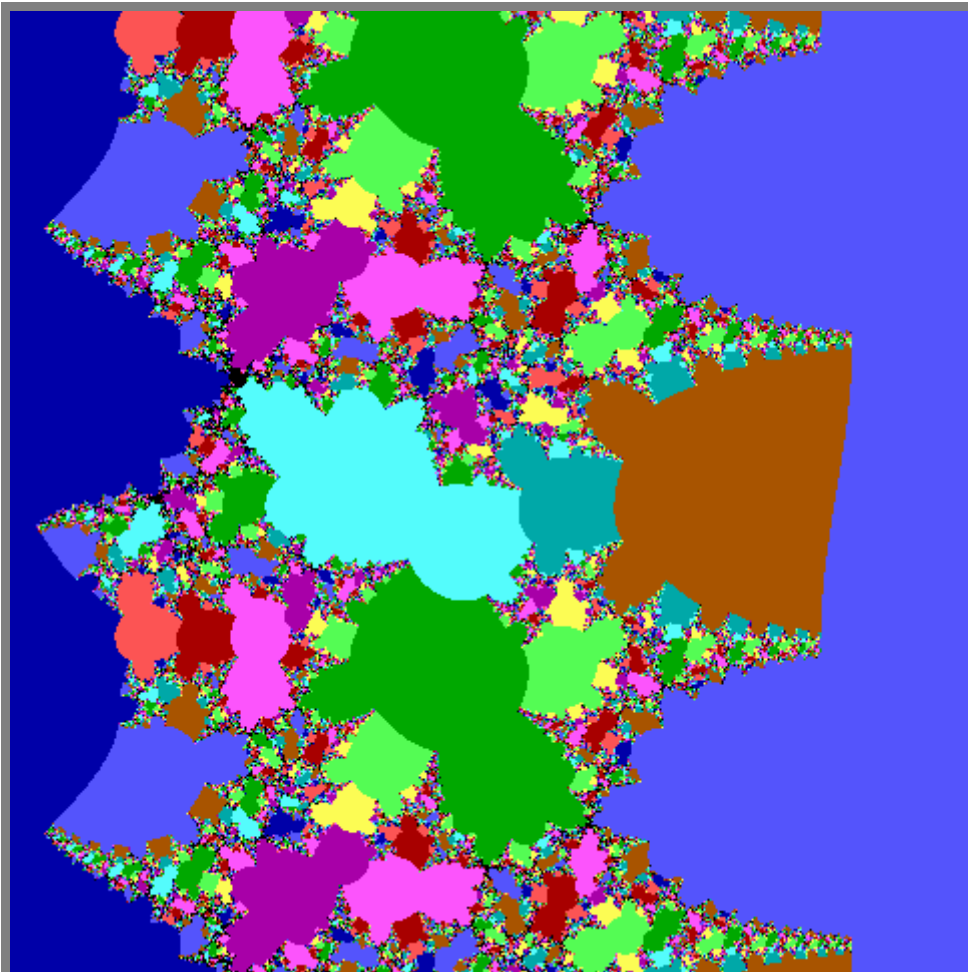




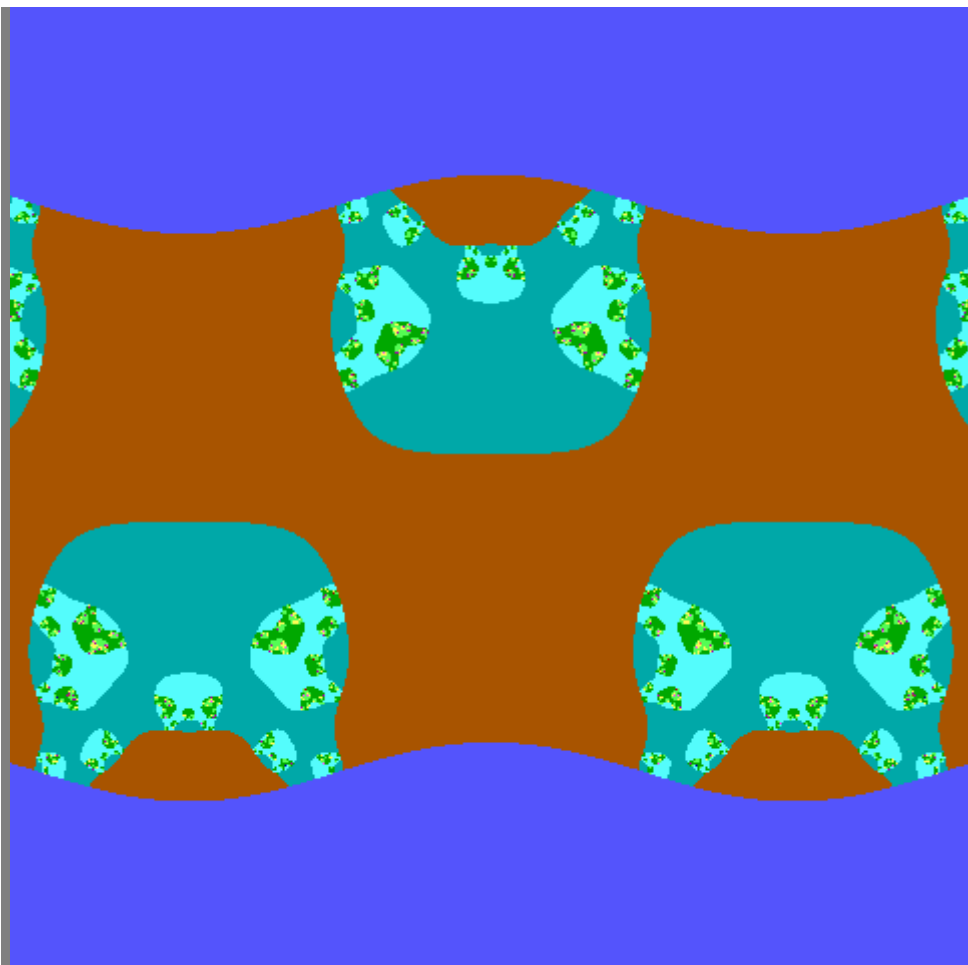


Если знать алгоритм раскраски, можно заметить, что множество, похоже, стало "нечестным". Чёрный цвет в нём не должен соседствовать более, чем с одним другим. Данное соседство свидетельствует либо о недостаточном разрешении на экране (то есть, там есть ещё "волоски" других цветов, но слишком короткие, чтобы их заметить), либо, что более вероятно, неправильно работает критерий расходимости. С научной точки зрения это значит, что мы обсчитываем уже не множество Жюлиа, а нечто другое. Но это другое по-прежнему достаточно интересно выглядит, и использование экспоненты определённо позволяет получать структуры, не проявляющиеся в обычном квадратичном отображении:





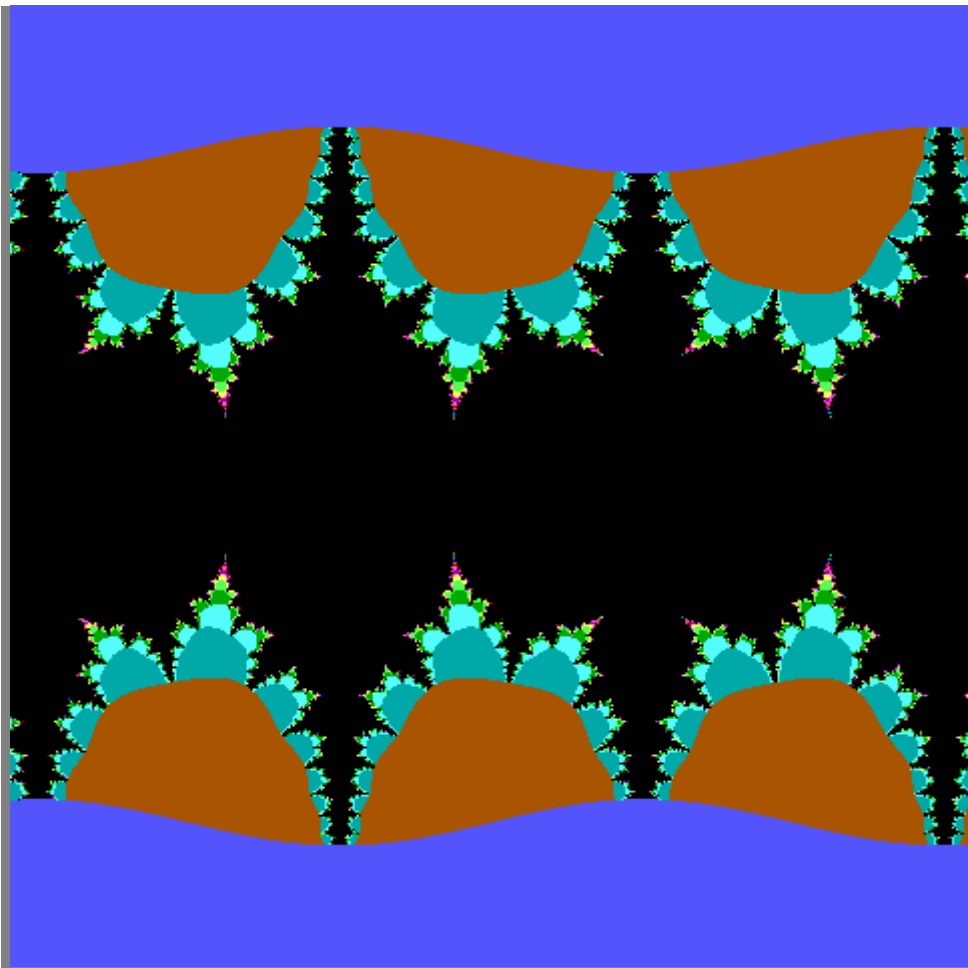
Ладно, закончили с экспонентой. Какие ещё у нас есть ряды Тэйлора? Ну конечно же, косинус:



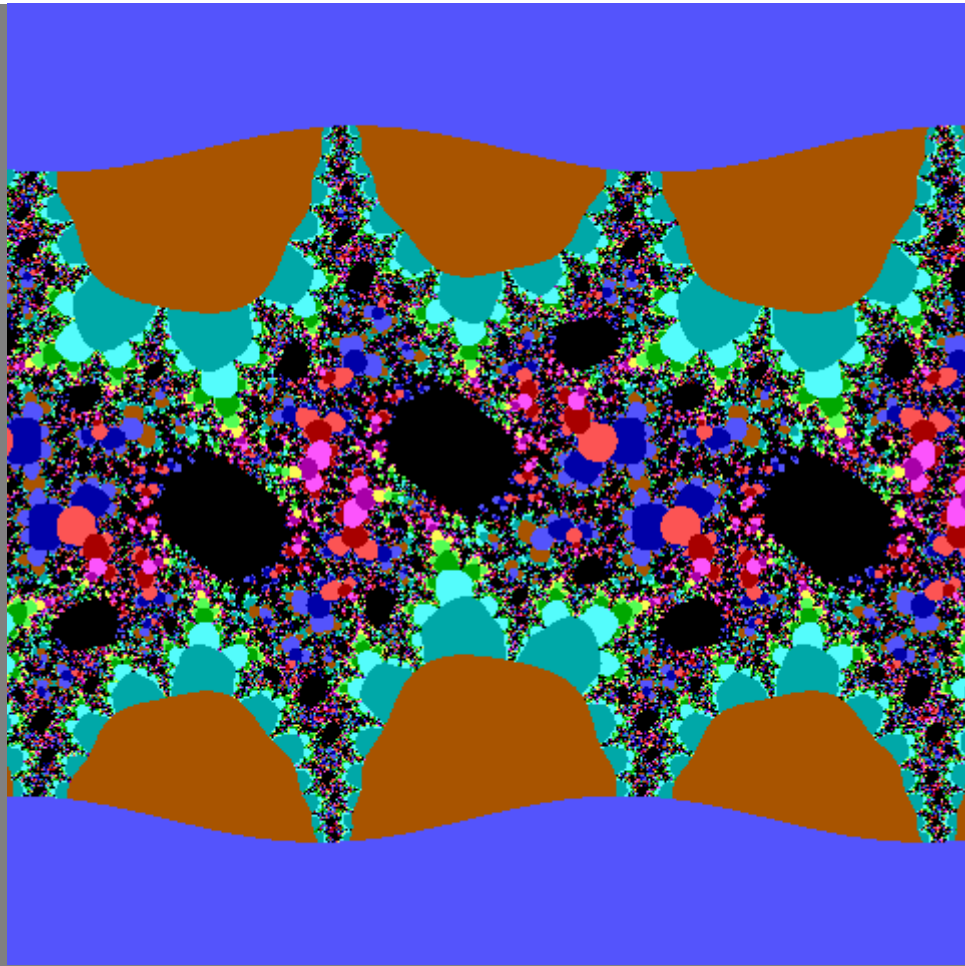
Константа равна  $3i$ .

Нисколько не удивительно: экспонента периодична вдоль мнимой оси, косинус -- вдоль действительной. Поскольку их ряды родственны, то и картинки получаются в некоторой степени похожими. Но косинусовские мне нравятся больше. Особенно если поиграть константами:

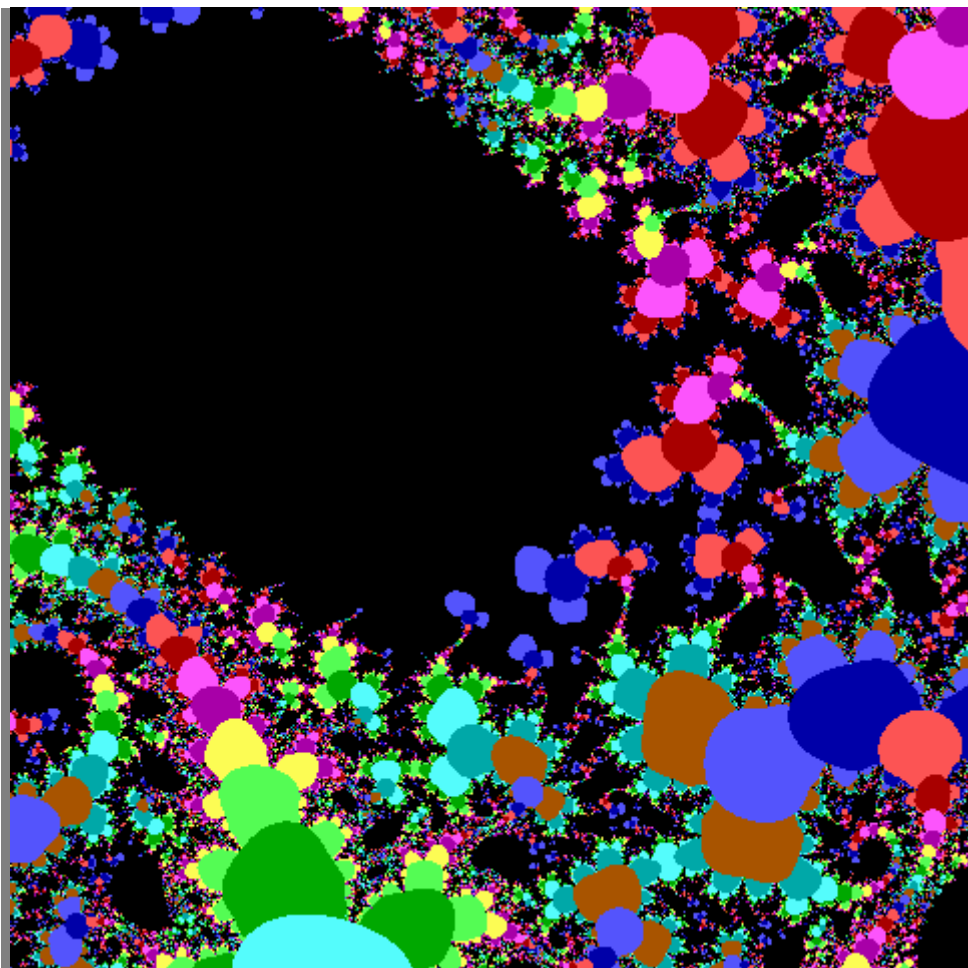
$c = 4.0$ :



$c = 3.9 + 0.9i$ , квадрат 5 на 5:

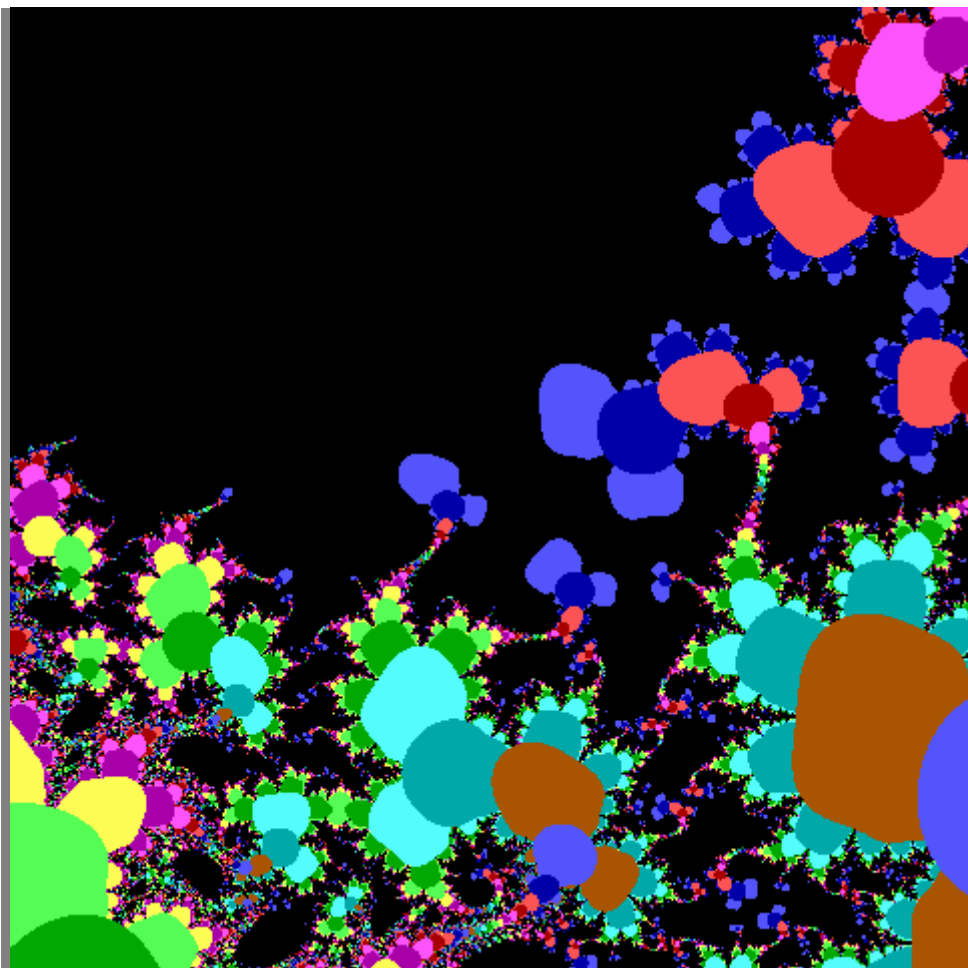


$c = 3.9 + 0.75i$ , квадрат 1 на 1:

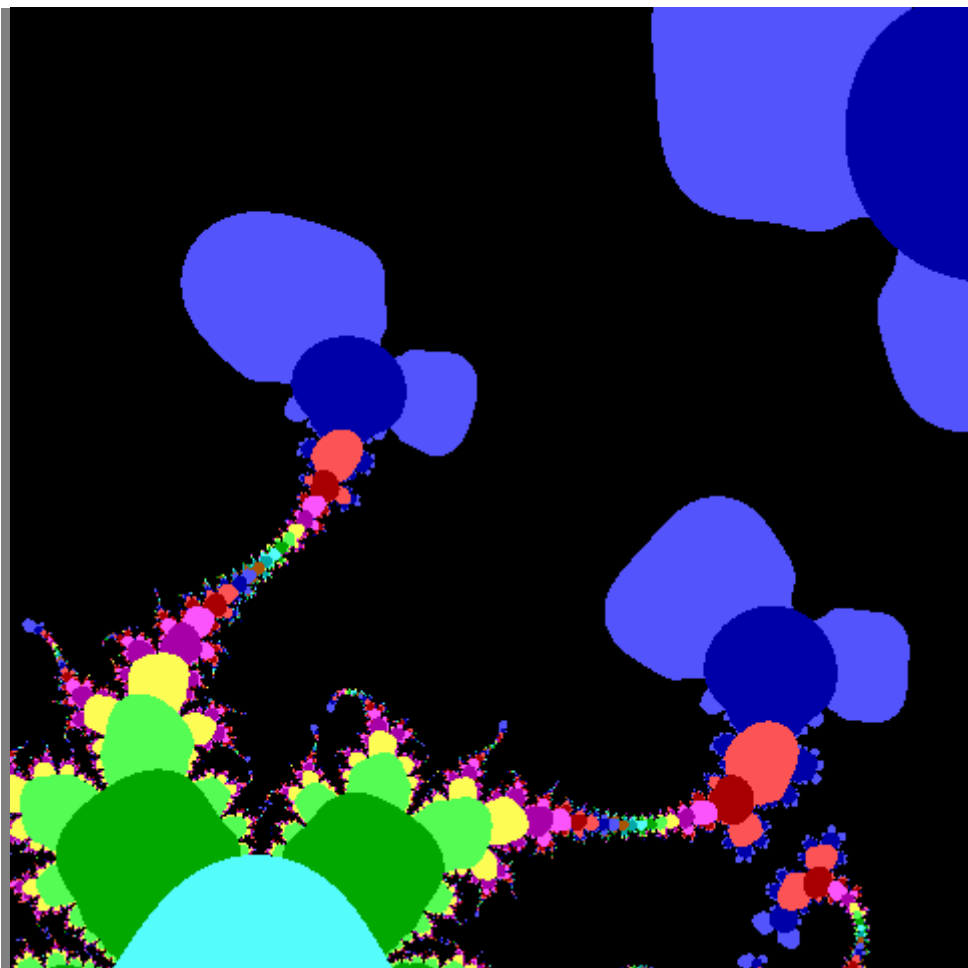


То же самое, но квадрат 0.5 на 0.5, смещённый относительно центра координат на  $+0.25i$ :



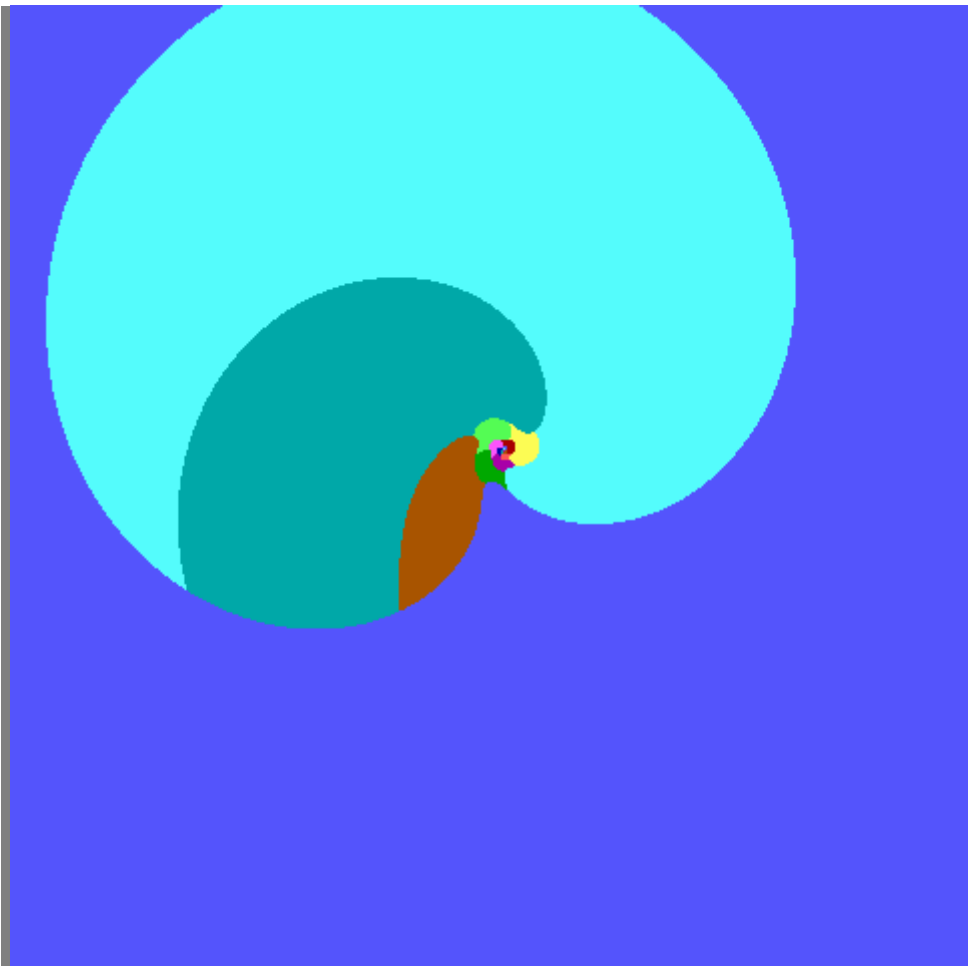


**0.15 на 0.15, сдвиг +0.30i:**



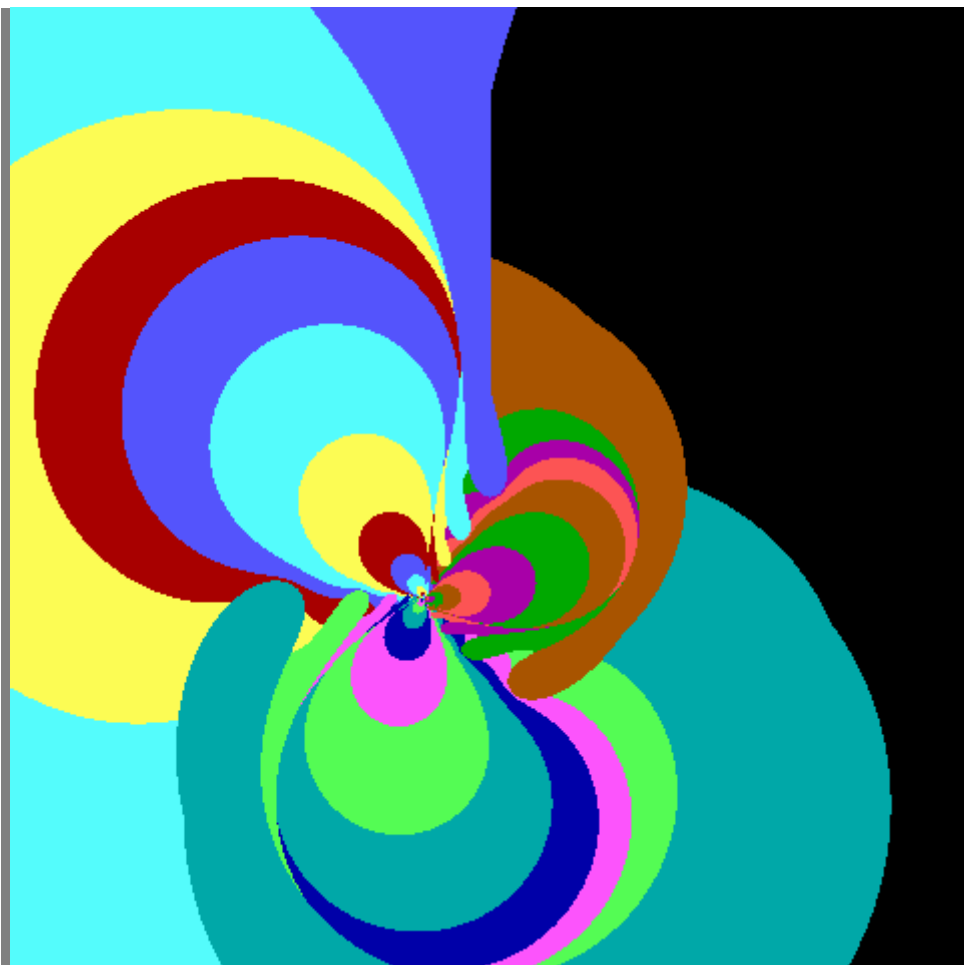
Всякий ряд Тэйлора -- это аналитическая функция. Какие ещё есть "стандартные" аналитические функции? Пробуем логарифм:

$Z_{n+1} = \text{Ln}(Z_n) + c$  при  $c = -1.2i$ , квадрат 10 на 10:



Не впечатляет... :( Ну то есть повторяемость наблюдается, но очень уж примитивная.

И как я ни крутил, ничего особо хорошего добиться не удалось. Вот это, пожалуй, лучшее, при том что понятно что обман:



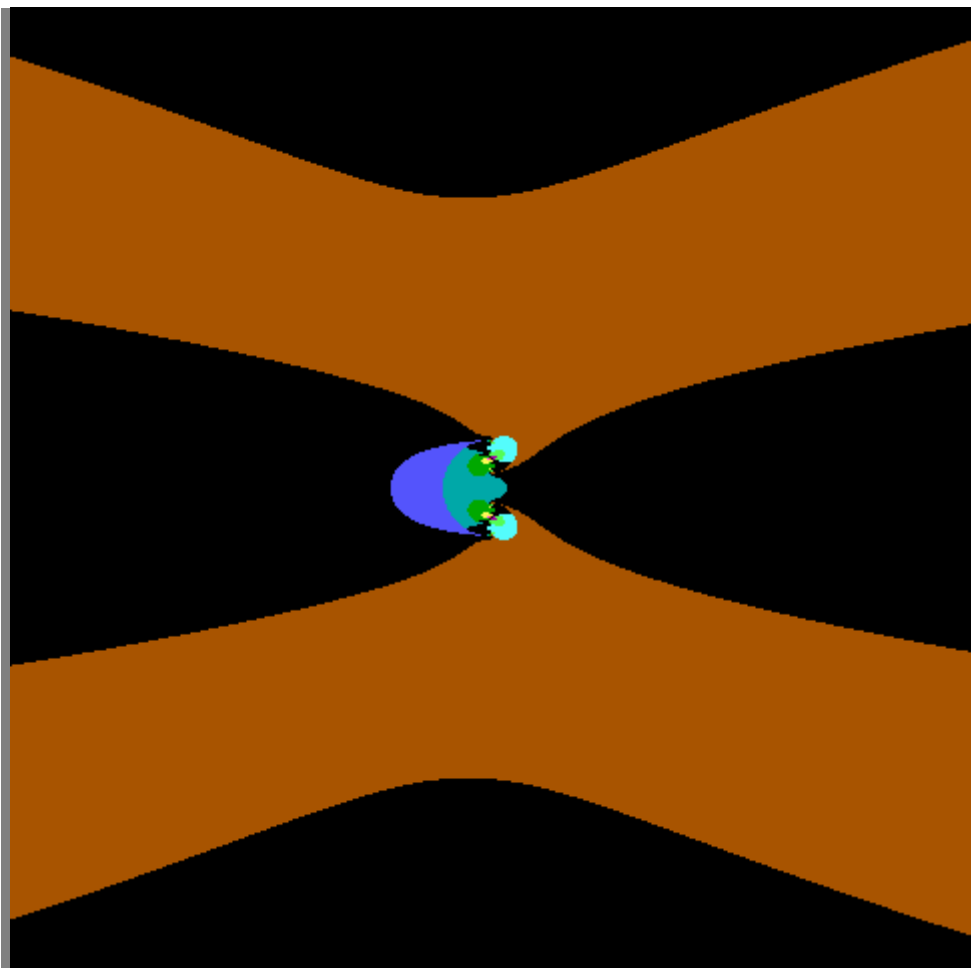
( $c = 0.2 - 1.428i$ , область  $3 \times 3$ ).

Гипотеза: картинку "портит" полюс в нуле?

Ладно, а что насчёт комплексного арктангенса?

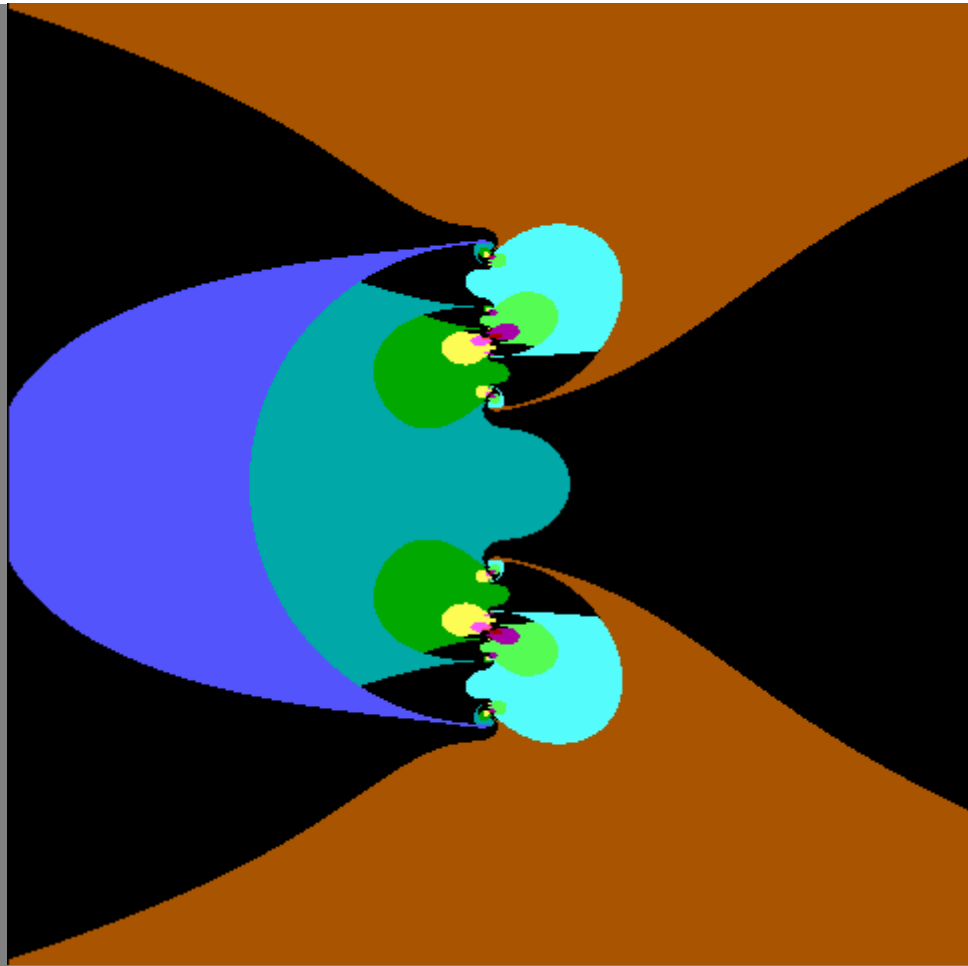
$$Z_{n+1} = \text{Arctg}(Z_n) + c?$$

Тоже не очень впечатляет ( $c = 1.15i$ ):

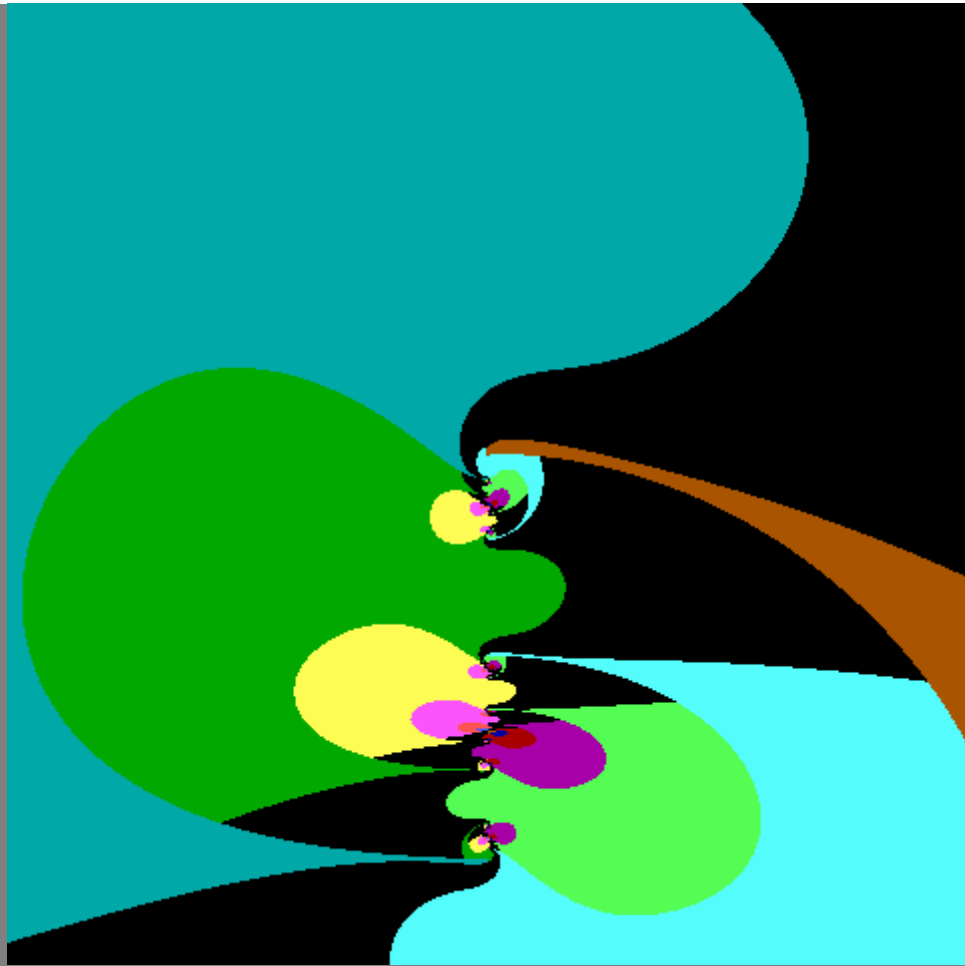


Похоже, опять мешают полюса, которые у арктангенса сидят в  $\pm i$  (нарисован квадрат 10 на 10). Но между этими полюсами, вероятно, должна же быть какая-то жизнь?

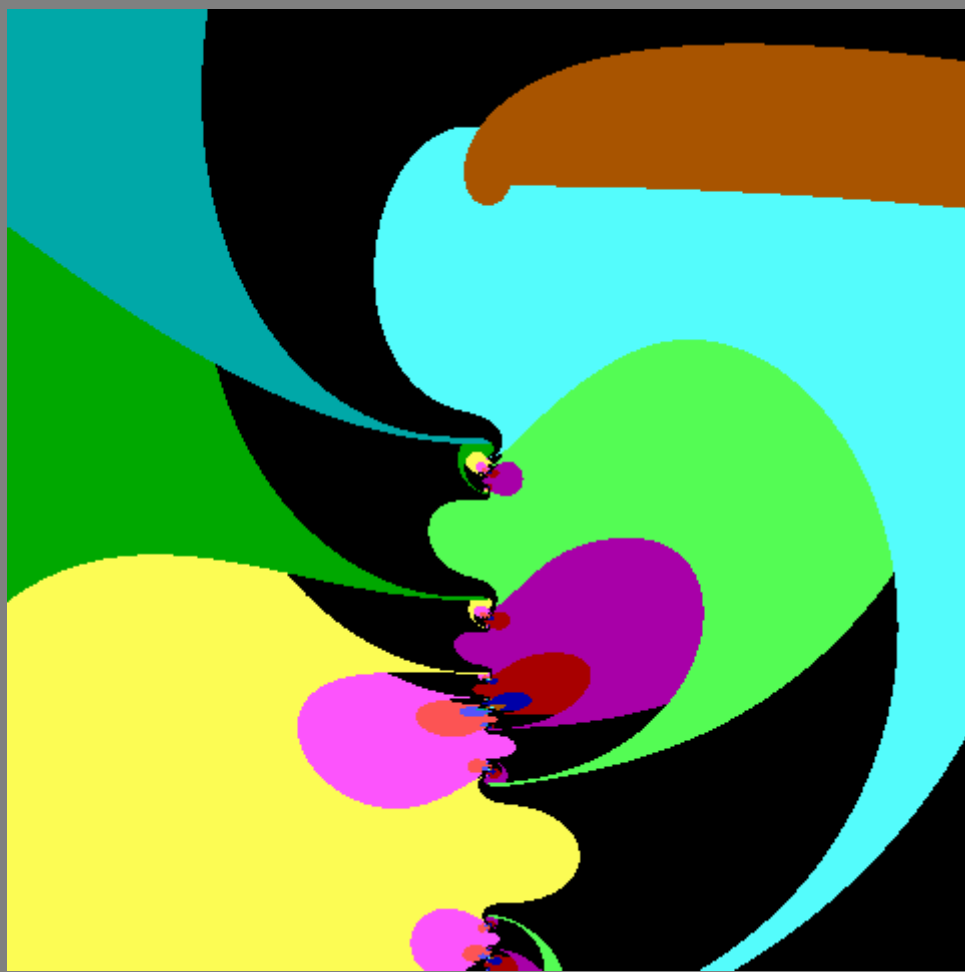
Последовательное увеличение подтверждает эту догадку:



Фрактал, причём весьма занятной структуры!



(Центр квадрата смещён вниз на  $0.35i$ )



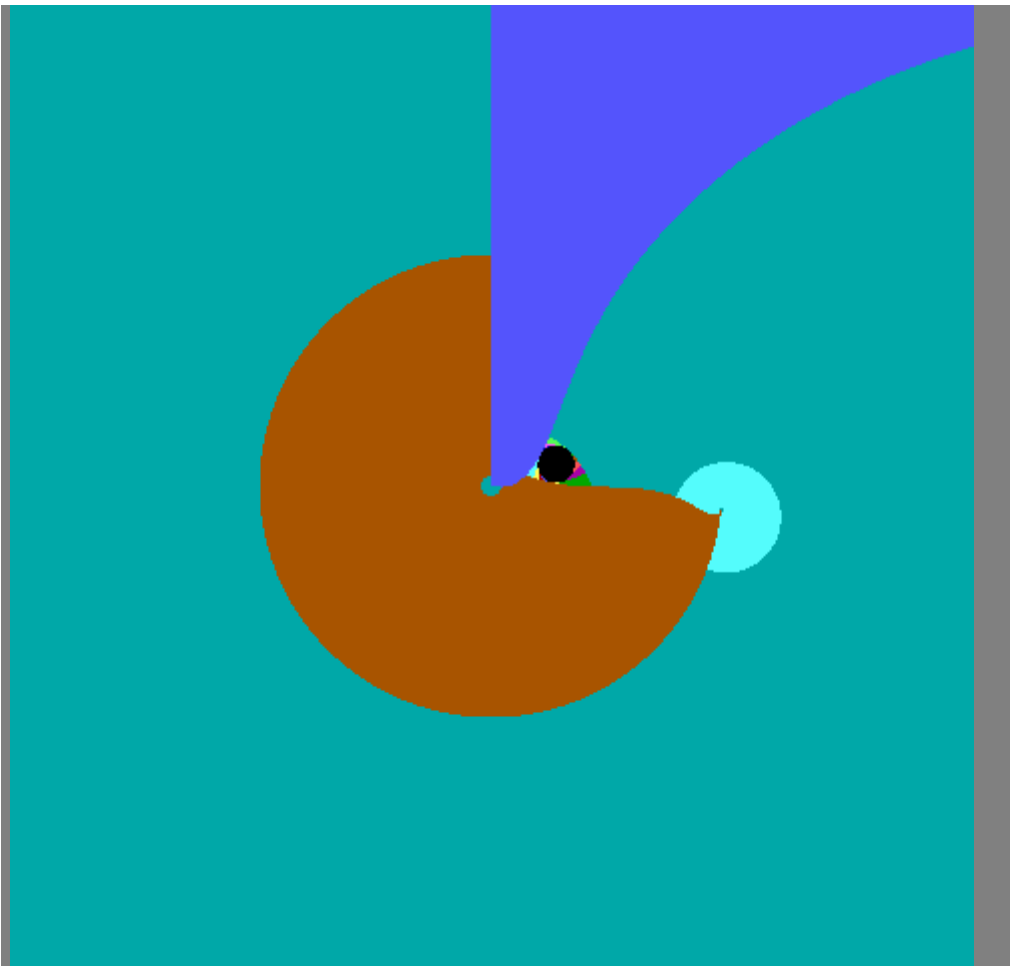
Итак, гипотеза: чем дальше полюса функции от отображаемой области либо константы, тем "живее" получаются картинки. Лучше всего -- когда все полюса на бесконечности, тогда вообще хорошо.

Дальше пошёл уже чистый математический беспредел; никакого внятного рационального обоснования этому нет. Что если попробовать опять степени... но комплексные?

Скажем,  $z_{n+1} = z_n^i + c$ ?

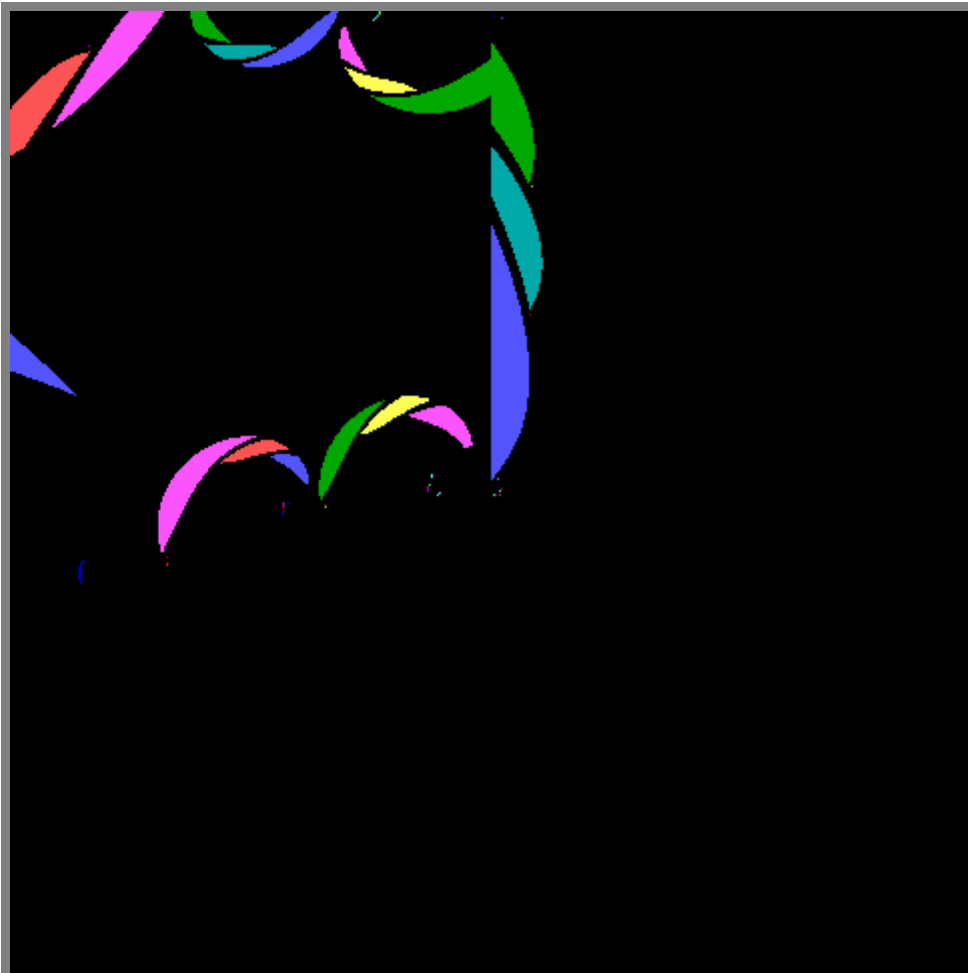
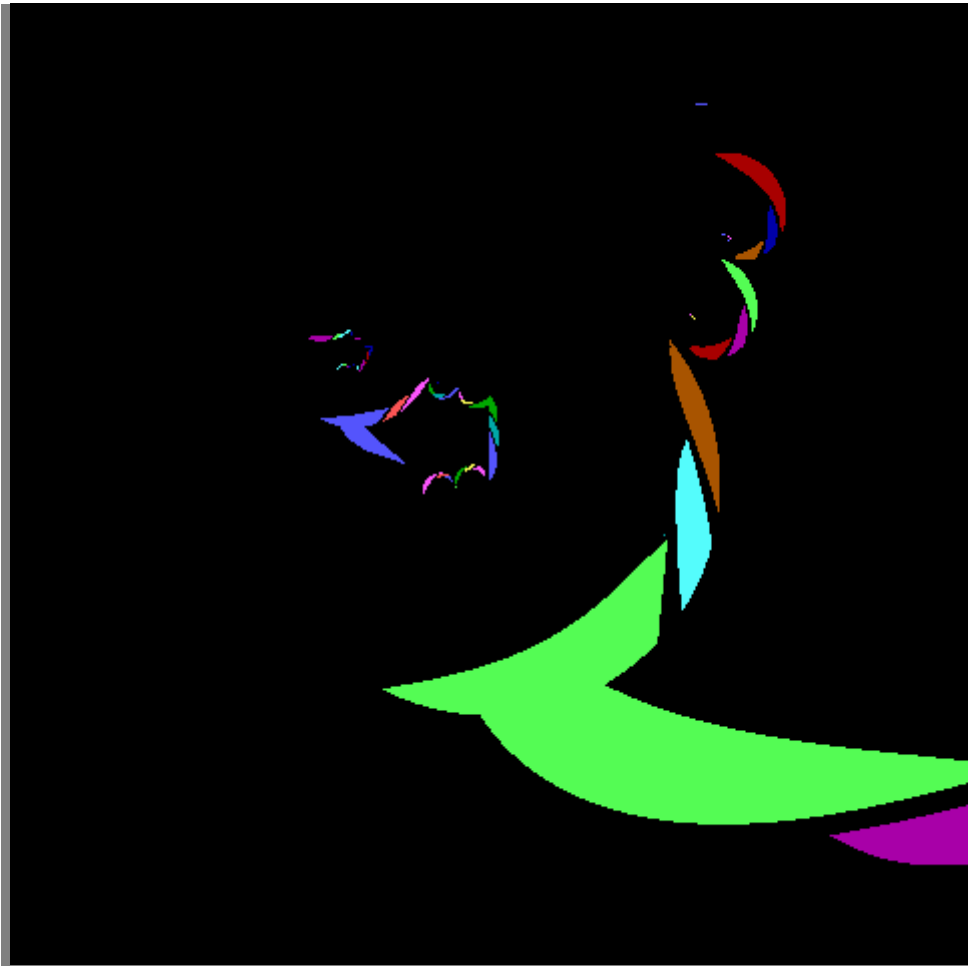
Как и ожидалось, влияние полюса в нуле оказывается довольно пагубным ( $c = -0.9i$ , квадрат 10 на 10):

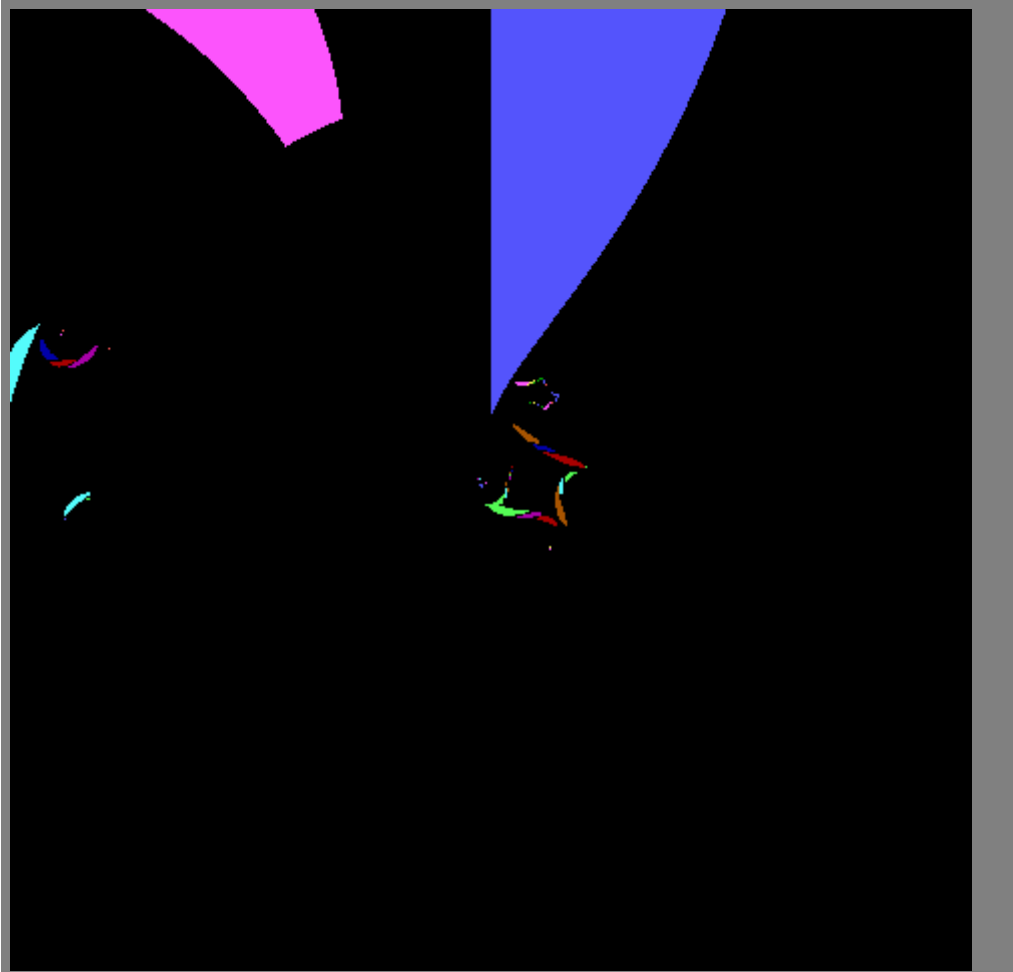


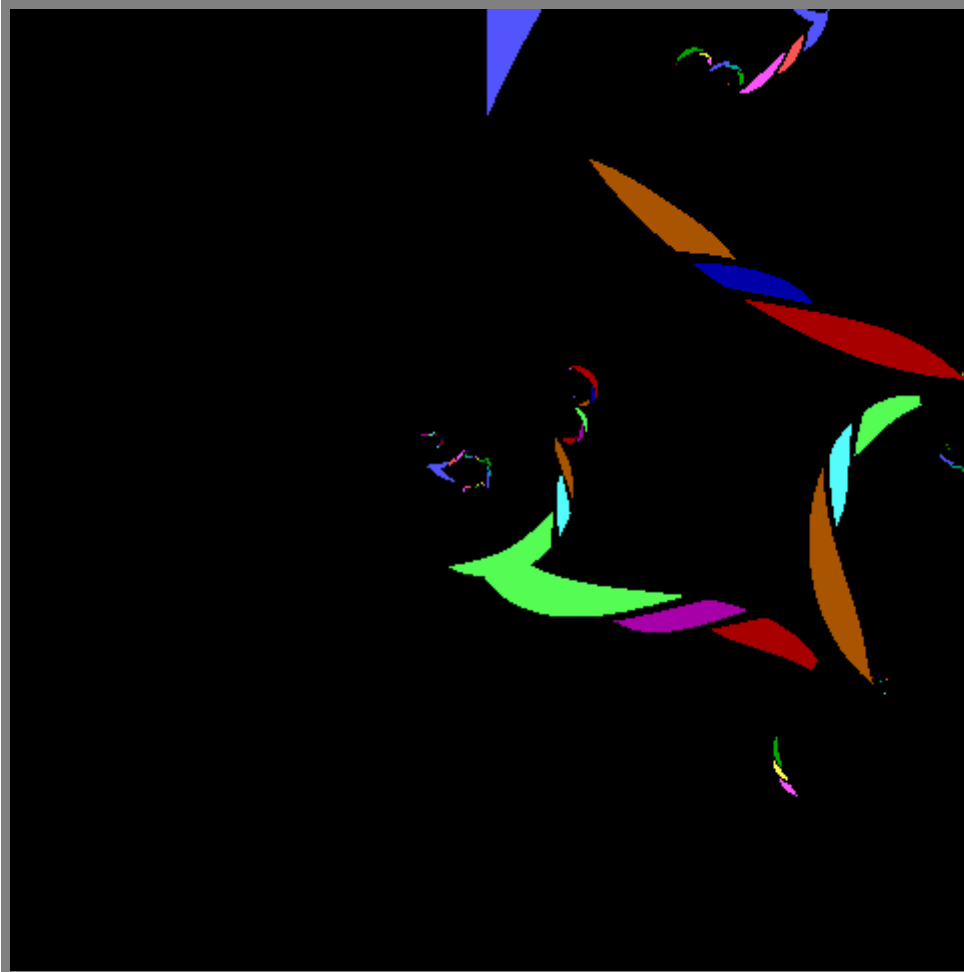


Вертикальная линия -- следствие того, что операция возведения в степень на комплексной плоскости, вообще говоря, неоднозначна для нецелых степеней. Где-то должен проходить разрез по фазе. Библиотека комплексных вычислений, которую я написал 15 лет назад, почему-то провела его по  $\pi/2$ . Зачем -- сегодня уже фиг вспомнишь; в более поздние университетские годы нас учили обычно проводить его вдоль нуля.

Слегка поиграв константами, эту картинку всё же можно сделать более интересной (правда, тоже, очевидно, нечестной). Видимо, помогает сложность полюса  $z^i$  в нуле:







Ну и на закуску, завершив круг, вернёмся к функции почти той же, с которой начинали:  $z_{n+1} = z_n^{2+i} + c$  при  $c = -0.01 - 4.4i$ , квадрат 10 на 10:



===

**Text Author(s):** Eugene Bobukh   ===   Web is volatile. Files are permanent. **Get a copy:** [[PDF](#)] [[Zipped HTML](#)]   ===   **Full list of texts:** <http://tung-sten.no-ip.com/Shelf/All.htm>]   ===   **All texts as a Zip archive:** <http://tung-sten.no-ip.com/Shelf/All.zip>] [mirror: <https://1drv.ms/u/s!AhyC4Qz62r5BhO9Xopn1yxWMSxtaOQ?e=b1KSiI>]   ===   **Contact the author:** h o t m a i l (switch name and domain) e u g e n e b o (dot) c o m   ===   **Support the author:** 1. **PayPal** to the address above; 2. **BTC:** 1DAptzi8J5qCaM45DueYXmAuiyGPG3pLbT; 3. **ETH:** 0xbDf6F8969674D05cb46ec75397a4F3B8581d8491; 4. **LTC:** LKtdnrau7Eb8wbRERasvJst6qGvTDPbHcN; 5. **XRP:** ranvPv13zqmUsQPgazwKkWCeEaYecjYxN7z   ===   **Visit other outlets:** Telegram channel <http://t.me/eugeneboList>, my site [www.bobukh.com](http://www.bobukh.com), Habr <https://habr.com/ru/users/eugenebo/posts/>, Medium <https://eugenebo.medium.com/>, Wordpress <http://eugenebo.wordpress.com/>, LinkedIn <https://www.linkedin.com/in/eugenebo>, ЖЖ <https://eugenebo.livejournal.com>, Facebook <https://www.facebook.com/EugeneBo>, SteemIt <https://steemit.com/@eugenebo>, MSDN Blog [https://docs.microsoft.com/en-us/archive/blogs/eugene\\_bobukh/](https://docs.microsoft.com/en-us/archive/blogs/eugene_bobukh/)   ===   **License:** Creative Commons BY-NC (no commercial use, retain this footer and attribute the author; otherwise, use as you want);   ===  
**RSA Public Key Token:** 33eda1770f509534.   ===   **Contact info** relevant as of 7/15/2022.

===