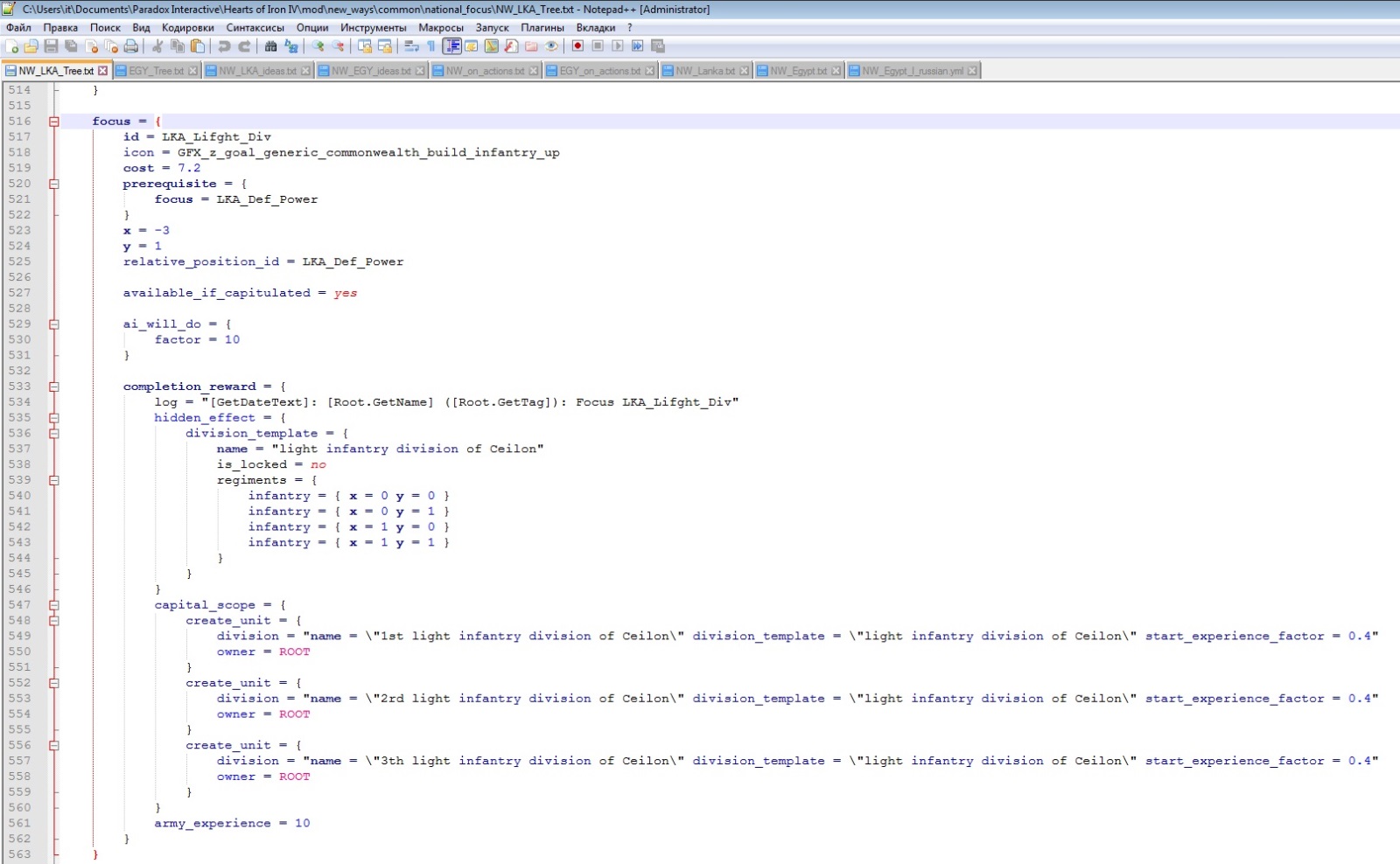
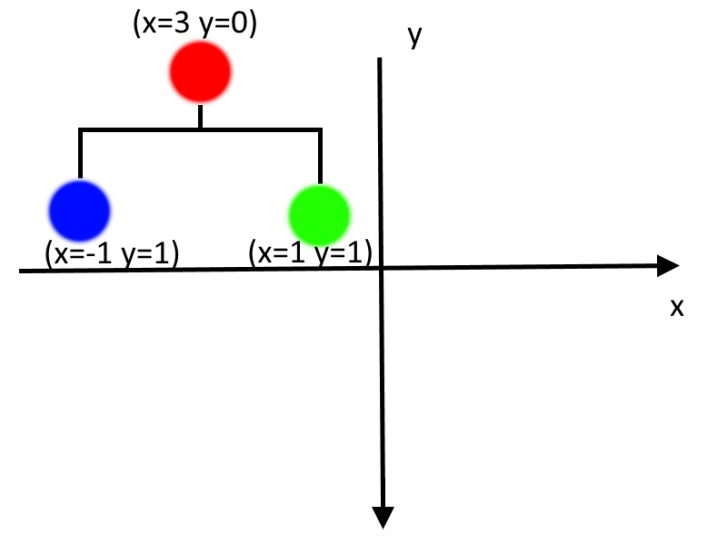
Поскольку у нас в команде несколько скриптеров, имеется небольшой список основных правил, которые, нужно соблюдать при написании скрипта. В случае ухода, или ошибки, которую необходимо будет экстренно исправить, другой скриптер должен суметь сориентироваться в вашем коде. Вот сам список:

**1. Соблюдать табуляцию.**



Суть простая. Как и показано на примере выше, ВНУТРИ каждой скобочки, нажимать TAB, делая отступ. Преимущества в этом два: вы потеряете нигде скобку(а это бывает даже у бывалых скриптеров), так в разы легче найти отдельный участок кода, отвечающий за что либо.

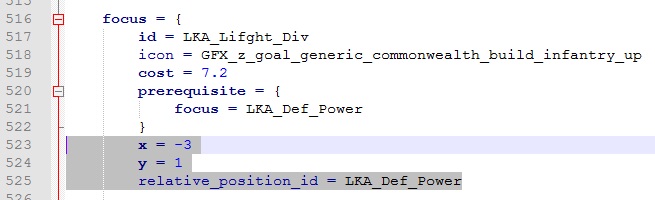
**2. Использовать релятивки.**

Для начала, что такое релятивки, и с чем их едят? Сначала может показаться, что это сложно, и зачем нужно? Но когда вы станете передвигать древо, или добавлять новые ветки, вы возблагодарите богов! Итак:  


Красный круг - фиксированная позиция. Два других - релятивка, относительно красного. При добавлении фокуса вниз, игрик прибавляется. При смещении фокуса влево — х уменьшается, а вправо — увеличивается. Надо раздвигать их координаты, смотреть, кто от чего зависит. Это так же можно смотреть в фтеме, там оно желтой линией отображено:



Двинешь верхний фокус к которому привязаны другие и вслед за ним поедет все дерево. Чтобы что-то вставить, надо учитывать координаты всех вышестоящих и нижестоящих фокусов. В коде это выглядит вот так:



**3. Фтем использовать только для создания СКЕЛЕТА дерева.**

Ну и просто просмотра в процессе работы. После составления дерева, его нельзя юзать, ибо код программы посредственный, и после пересохранения может создать вам только кучу багов.

**4. Стараться сразу оптимизировать код.**

Наш мод довольно большой, и растёт с каждым днём, а значит, мы перешагнули тот этап, когда оптимизация была не особо важна. Если вкратце о том, что как минимум необходимо: В решениях для родной страны прописывать ROOT, а не её тег, и желательно создавать решения не каждое самостоятельно, а одной категорией, это так же оптимизирует код. В программном коде максимально использовать ROOT и FROM. Если более подробно, то об оптимизации можно почитать тут(от лёгкого к сложному):

<https://vk.com/@heartsofmodding-1-glava-optimizaciya-modifikacii-osnovy>

<https://vk.com/@heartsofmodding-2-glava-optimizaciya-modifikacii-uskorenie-cherez-resheniya>

<https://vk.com/@hoi4nw-kak-uprostit-zhizn-skripteru>

<https://vk.com/@-187801996-kak-napisat-horoshii-kod-gaid-na-russkom>

**5. Делать логификацию эффектов.**

Логи – самое просто средство для обнаружения ошибок в процессах игры. Оно пишется в самом начале получения награды. Ниже будут несколько примеров.

Для Фокусов:

completion\_reward = {

log = "[GetDateText]: [Root.GetName] ([Root.GetTag]): Focus LKA\_Irrac\_Build"

add\_ideas = LKA\_idea\_irr\_building

}

Для ивентов:

country\_event = {

id = lka.13

title = lka.13.t

desc = lka.13.d

picture = GFX\_report\_event\_lka\_37

is\_triggered\_only = yes

immediate = { log = "[GetDateText]: [Root.GetName] ([Root.GetTag]): event lka.13"}

option = {

name = lka.13.a

add\_popularity = { ideology = trozkism popularity = 0.03 }

add\_stability = -0.05

}

}

Для решений:  
remove\_effect = {

log = "[GetDateText]: [Root.GetName] ([Root.GetTag]): remove decision LKA\_island\_rises\_2"

set\_country\_flag = LKA\_dis\_13

}