



## Volume 2 – O Alelo Perdido

“*Queria Star morta*”, pensou a tenente ao chegar a anomalia temporal. Cansada, confusa e meio sem entender direito o que tava acontecendo naquela distorção do espaço-tempo, ela respirou fundo, buscou energias e coragem dentro de si e foi. Apenas foi.

“Warray, Relatório, por favor.”

“Tenente, chegamos. Estamos a poucos minutos de Oya. Já liguei nossos sensores e tenho uma leitura parcial. A nave **Curious XXIV** está orbitando o planeta Oya. O meu acesso aos computadores já terminou e já fiz o download do Banco de Dados da Nave e dos veículos de exploração em solo. Mas... Tenente... Nenhuma forma de leitura indica vida na nave... E nenhum drone ou rover em terra mostra contato recente da nave.”

Apesar de já esperar por isso, a tenente da missão sentiu uma angústia profunda. Mas maior que sua angústia foi a certeza de que **ela não poderia parar**. Que usaria **todo** o conhecimento dela para desvendar esse mistério e honrar aqueles pioneiros exploradores e suas famílias.

\*\*\*

Agora tenente, é com você.

Nessa etapa da Missão, você terá uma série de **achievements**. Afinal, estar em um local desconhecido da galáxia requer sangue no olho; Mas isso não é problema, afinal, você tem.

Os efeitos gravitacionais alteraram algumas funções dos circuitos do Warray, então você vai ter que reprogramá-lo, por meio de um processo que chamamos de **Refatoração**. Essa parte será explicada depois, agora, vamos focar na **busca pelo(a) assassina(o)**

Como dito na **Carta do Império**, há uma suspeita de assassinato. Você deve ir ao **SimGem** e processar os **Fragmentos Cromossômicos** enviados por amostra (*Sample*) pela **Curious XXIV**.

Você deve **criar um programa em linguagem C** que seja apto, ao final, de dizer quais as características principais da **amostra** e, baseado nelas, a **forma** do possível Ser.

//Nota Técnica: Quando a tenente escolher a opção **SimGem** no menu principal do Wararray, deve ser chamada um função, que deve ter o nome **start\_SimGene()**;

Cada arquivo contem um grande número de caracteres. Você deve passar esses caracteres para a função.

Deles, você deve descobrir **qual a probabilidade de expressão de um gene**. Um gene se “expressa” sempre que ele está presente. Mas um indivíduo tem vários genes.

E como saber o que é um gene e qual gene codifica o que?

A Ph.D Gastd Moyses, chefe-mestra da Divisão Genética da Nave **Curiosity XXVI**, documentou as pesquisas e achados. Ela classificou os tipos de genes e correlacionou eles a características dos indivíduos. **Leia o Diário de Pesquisa** dela para buscar todas as informações necessárias.

Mas, caso você já tenha lido antes de abrir esse arquivo... você verá que há ruído, muito ruído na informação. Parte dela corrompida. Mas, por milagre de **Oya**, as fitas são complementares! Ou seja, tudo que falta em uma, tem na outra!

Agora, sabendo que o **DNA** conserva a característica **Universal**, lembre quais bases ligam-se com quais. E para mais dúvidas, consulte o **Diário de Pesquisa**.

Então, tendo em mente que você precisa **contar** os tipos de genes, calcular a probabilidade dado a ocorrência e daí classificar o fragmento de DNA, releia conhecimentos sobre **Espaço Amostral e Probabilidade de Eventos Independentes**.

Você viu que vai ser um trabalho que requer atenção, não é? Então, para que você possa usar o SimGene, a nave deve estar com o **Piloto Automático ligado**. A tenente da nave pode ligar o Piloto Automático no menu ou, ao acessar o SimGene, ele pode ligar o piloto automático pra você, caso você permita.

//Nota Técnica: apenas 1 única variável deve ser usada para dizer se o piloto automático está ligado ou não, ou seja, não adianta em cada função dizer, em uma variável qualquer, que ele está ligado e usar duas, três ou mais variáveis diferentes pra testar em um condicional. Fazer isso implica em **erro de lógica e implementação**.

Quando o SimGene() terminar a Análise de qualquer um dos arquivos, ele deve mostrar em tela todas as informações relevantes para que quem leia possa ter conteúdo, organizado e claro suficientes, para entender o fragmento.

//Nota Técnica: **sugestão**: colocar quanto (em termos absolutos) de cada gene foi encontrado e as análises de porcentagem. Mas apenas isso é suficiente para uma boa leitura? O que mais poderia ser relevante?

Agora, sabendo analisar e entender todos os genes e calcular suas probabilidades, você deve, **em um documento à parte**, responder a maior questão: qual das amostras você diria ao Império Gláctico que é responsável pela carnificina? Que evidências levaram você a essa conclusão?

## Refatoração

Essa parte é completamente optativa. Mas lembre-se: cuidar bem do seu **android** fará com que ele **cuide bem de você**.

Atualize Wararray:

[ ] Na parte de cálculo do Salto em Hiper-Espaço, ponha loops e tratamento de erro nas entradas.

[ ] Transforme todo o trabalho desenvolvido na **main()** em uma **função à parte**.

[ ] Crie uma opção para o Salto em Hiper-Espaço que agora leva em consideração apenas o módulo. Considere que você é sempre o ponto Xo, Yo, Zo. Considere que os sensores da nave conseguem ler a posição de objetos que estão até **24 mil Whormholes de distância**. E, por fim, troque o maquinário da sua nave pelo da **Curiosity XXIV**, assim atualize as fórmulas:

[ ] Cada motor tem um **rendimento** de **mil Whormholes (WH)** por **Combustion(CN)**.

[ ] Pra sua nave ficar mais leve, você trocou por motores de liga de Aminostácio Hergulano (Muito melhor que aqueles de Permita Puriliana ridículo), ou seja, seus motores tem um peso desprezível para a potência que eles empregam. Assim, a partir de agora, a distância que você percorre só depende de **Quantos Motores** você tem (Máximo de 6); Quanto **Cada Um** deles tem de **Combustível** (Máximo de 66 Combustions). E como os **Eventos Intergalácticos afetam** o rendimento da Nave. Assim, o cálculo que o Wararray fará será bem mais simples: Rendimento Atual é o 100% que ele teria da teoria menos o tanto que cada evento afetou ela. Se você pode ler a quantidade de combustível inicial de cada motor, você sabe quanto que cada motor vai impulsionar a nave e, assim, quanto ela vai poder percorrer! (=

[ ] Salve as seguintes informações para imprimir na tela:

[ ] Em um Array de Char, Planetas/Lugares que você esteve;

[ ] Em um Array de tipo adequado, a quantidade de combustível em cada Motor antes de cada Salto;

[ ] Em um Array de tipo adequado, a eficiência Atual da Nave antes de cada Salto;

[ ] Faça com que o Wararray possa interagir com o sistema de Guerra da nave:

[ ] Crie 3 tipos de escudo contra 3 coisas que você ache importante. Escudos reduzem dano a eficiência e a algo que chamaremos de **Avarias**. A avaria é uma variável que, quando chegar em 100% significa que sua Nave está **Morta**.

[ ] Crie 3 tipos de armas de ataque que causam dano ou ao **rendimento**, e/ou a **velocidade** e/ou afete a avaria do objeto/obstáculo/oponente que for acertado por ela.

[ ] Warray só pode permitir a Nave ter 1 tipo de Escudo e 1 tipo de Arma ligado ao mesmo tempo;

[ ] Salve todas essas informações de uma maneira que fique fácil recuperá-las quando for necessário, seja pra alterar ou consultar ou criar um relatório;

[ ] Todas as informações, quando não fizerem sentido (como combustível negativo, motores decimais, etc.) devem ser tratadas pelo Warray. O que fazer quando isso acontecer? Use seu intinto racional e lógico para ensinar Warray a tomar iniciativas que julgar adequadas.

[ ] Todo planeta/estrutura/criatura/etc que você criar, **deve ser explicado e documentado** no nosso **Guia do Mochileiro das Galáxias Para Navegantes**. E todo caminho que você percorrer, todo lugar que você visitar, **deve ser colocado no seu Diário de Bordo**.

Toda refatoração só será considerada para avaliação se estiver bem documentada e explicada. O Comando Gláctico leva a sério todo bom Relatório de Todo bom e Toda boa Tenente. Organize-se. Deixe claro procedimentos. Só depende de você saber comunicar as coisas de maneira clara e eficiente.

## REGRAS ESPECÍFICAS DA ESTRUTURA DO TRABALHO:

☐ **Toda e qualquer** entrada do trabalho deve ser tratada. Não vai adiantar escrever no ReadMe ou instruir ao usuário “Tecle somente X ou Y”. A tenente acabou de saltar em Hiper-Espaço. Elá tá L0ka, miga. Não vai digitar certo. **Trate Todas as Entradas**.

☐ Em todas as **estruturas do Volume II, simGene()**, o usuário vai poder **errar quantas vezes quiser**. O programa só poderá sair se ele quiser ou se terminar a execução, mostrando algum resultado.

## REGRAS ESPECÍFICAS DA MORFOLOGIA DO TRABALHO:

- ☐ Variáveis Mnemônicas são obrigatórias;
- ☐ Identação precisa ser obedecida;
- ☐ Código deve ser comentado/auto-comentado;
- ☐ toda nova função deve ter o **escopo** definido **antes** da main() e o conteúdo/lógica deve vir **depois** da main();

## REGRAS ESPECÍFICAS PARA ENTREGA DO TRABALHO:

☐ ReadMe: Aqui você vai explicar a lógica que você usou, como você pensou, como compilar, identificar e analisar tudo o que for necessário para justificar o que você fez no seu trabalho. Vamos dar alguns exemplos de bons arquivos ReadMe pra vocês.

☐ O título do arquivo .c deve ser: MATRICULA\_NOME. Por exemplo: 120044749\_MonitorSarado.c . A PASTA que você compactar deve ser seu Nome e Sobrenome.

☐ Identifique TODOS os arquivos mandados.

## REGRAS GERAIS DA ESTRUTURA DA LINGUAGEM:

- ☐ Fica vetado o uso das seguintes estruturas fornecidas pela linguagem C:
  - ☐ **Go To;**
  - ☐ **Switch Case;**
  - ☐ **Break;**
  - ☐ **Define;**
  - ☐ **Variáveis Globais;**
  - ☐ **Qualquer biblioteca não original de manipulação de strings (exemplo: string.h).**

## REGRAS ÉTICAS GERAIS:

- ☐ Plágio é **Zero**. E **Crime!**. Vlw, Flws.
- ☐ O trabalho é individual.
- ☐ **PROCURE AJUDA!** Monitores e Professores estão aqui para você! <3