

CIÊNCIA DA COMPUTAÇÃO

APS – ATIVIDADES PRÁTICAS SUPERVISIONADAS **JOGO VALLEN**

Gustavo Teixeira de Lima - G873270

Kaique Silva Pinto - F353028

Luan de Carvalho Leite - R004937

Luiz Gustavo do Prado Udorisse - R013421

Maria Eduarda Rodrigues Batista Ribeiro dos Santos - F3561C6

Northon Dias Conceição - R022FD7

Rafael Henrique Godinho de carvalho - G8731E2

Thiago Henrique Varano Evangelista - G85CEA5

Willyam Oliveira de Lima - R022FF3

Universidade Paulista

27/05/2024

ÍNDICE

OBJETIVO:.....	3
INTRODUÇÃO:	4
PLANO DE DESENVOLVIMENTO DO JOGO:	6
REFERENCIAL TEÓRICO:	11
Classe e Método main.....	11
Encapsulamento.....	11
Métodos Construtores	12
Getters e Setters	12
Criação de Objeto	14
Scanner.....	14
If e Else	15
Herança.....	16
REGRAS E FUNCIONAMENTO DO JOGO:	18
Criação de personagem:	18
Componentes do Jogo:	18
Ações do Jogador:	19
Cooperação:.....	20
Resolução de Conflitos:	20
Condições de Vitória:	21
Regras de Funcionamento:	21
ESTRUTURA DO PROGRAMA:	22
RELATÓRIO DO CÓDIGO:.....	26
PROGRAMA EM FUNCIONAMENTO:.....	45
BIBLIOGRAFIA:.....	46
FICHAS APS:	46

OBJETIVO:

O Presente documento descreve a criação do projeto de APS (atividades práticas supervisionadas), que têm como objetivo a criação de um jogo textual, desenvolvido na linguagem de programação orientada a objeto JAVA, visando testar os conhecimentos adquirido durante as aulas.

INTRODUÇÃO:

Vallen é um jogo de RPG (Role playing game) textual baseado em escolhas, cujo tema central é a preservação e educação ambiental. As personagens principais possuem atributos básicos (vida, força e agilidade) que podem ser aprimorados ao longo do jogo, conforme as decisões tomadas pelo jogador. As escolhas feitas determinam o rumo da história em cada situação, e o jogo oferece diferentes classes para ajudar o jogador a enfrentar o vilão.

No universo de Vallen, um cometa cai na Terra, transformando uma floresta em um santuário repleto de cristais celestiais. Uma tribo nômade decide se estabelecer ali, protegendo os cristais que garantem harmonia e prosperidade da floresta. Você é um jovem morador da aldeia de Vallen. Quando o antagonista, um cientista ambicioso, rouba um dos cristais para aumentar sua força e destruir a floresta, você descobre suas intenções sombrias. Junto com sua amiga, você deve investigar, enfrentar vilão e proteger os cristais para harmonia da floresta. As escolhas que você faz ao longo da jornada – desde investigar ou ignorar pistas até decidir como enfrentar o malfeitor – determinarão o destino da floresta mágica e da sua comunidade. Treinado pelo mestre ancião, você deve liderar os aldeões em uma batalha épica para salvar Vallen da ambição do vilão. Suas decisões definirão se Vallen prosperará ou cairá sob a tirania do adversário destruindo a preservação da floresta, “o futuro de Vallen está em suas mãos, faça sua escolha, jogador!”.

Em Vallen, os personagens interagem de duas maneiras distintas: através de diálogos e combates textuais. Durante as interações de diálogo, os personagens seguem padrões de fala predefinidos, refletindo suas personalidades, motivações e o contexto da situação. Esses diálogos permitem que os jogadores explorem a história, tomem decisões importantes e desenvolvam relacionamentos com os outros habitantes da aldeia.

O feedback das escolhas textuais é imediato e narrativo. Após cada decisão, o jogador recebe uma descrição detalhada das consequências de suas ações. Por exemplo: "Você investiga algo suspeito e descobre que realmente está acontecendo algo grave e procura compartilhar com alguém", "Sua tentativa de negociação falha, e você é atingido por um golpe, sentindo a dor atravessar seu corpo". Os resultados

das decisões podem variar de acordo com a situação específica e o estado do personagem, incluindo saúde, moral e recursos disponíveis. Além disso, a mecânica de combate utiliza uma lógica baseada em matemática simples. Nesse sistema, a força do jogador e de seus aliados é comparada à do malfeitor em uma análise direta. Se a força combinada do jogador e seus aliados superar a do oponente, a vitória é alcançada; caso contrário, a derrota é inevitável.

Nesse universo narrativo, a mecânica de combate textual não apenas influencia o desenrolar da trama, mas também determina os desfechos possíveis do jogo. Com base nas escolhas feitas pelo jogador ao longo da jornada, existem três finais distintos predefinidos, cada um representando um desfecho único e impactante. Esses desfechos finais são o resultado direto das decisões tomadas pelo jogador em momentos cruciais, refletindo não apenas suas habilidades estratégicas durante o combate, mas também suas escolhas morais e éticas ao longo do jogo. Essa estrutura de múltiplos finais adiciona uma camada adicional de complexidade à narrativa, incentivando os jogadores a explorar diferentes caminhos e enfrentar as consequências de suas ações. Assim, o jogo oferece uma experiência dinâmica e personalizada, na qual cada jogador pode moldar o destino do mundo de acordo com suas próprias escolhas e valores.

PLANO DE DESENVOLVIMENTO DO JOGO:

O desenvolvimento do projeto "Vallen" foi distribuído pela equipe de forma:

Maria Eduarda e Gustavo Teixeira

Discussão de ideias 15h

Desenvolvimento organograma 10h

Codificação 30h

Documentação 20h

Kaique Silva, Northon Dias, Willyam Oliveira e Luiz Gustavo

Discussão de ideias 15h

Pesquisa Codificação 20h

Codificação 40h

Luan de Carvalho, Thiago Henrique e Rafael Henrique

Discussão de ideias 15h

Pesquisa Codificação 10h

Codificação 20h

Documentação 30h

A criação do jogo foi fundamentada em duas principais ideias. O Eclipse foi utilizado para a codificação, proporcionando um ambiente robusto para o desenvolvimento e depuração do código-fonte. Paralelamente, o Figma foi empregado para a criação de um organograma, oferecendo uma organização de pensamentos e estrutura que deu suporte a desenvolver as ideias com clareza.

Cada membro da equipe desempenhou um papel fundamental no desenvolvimento do jogo, contribuindo com suas habilidades e conhecimentos específicos para garantir o sucesso do projeto. O desenvolvimento do jogo seguiu um cronograma detalhado, que incluiu as seguintes etapas:

INTRODUÇÃO

No princípio, um cometa caiu na terra, transformando uma antiga floresta em seu santuário e espalhando seus cristais mágicos em sete direções do lugar. Diante desse espetáculo celestial, uma tribo de nômades, tocada pela grandiosidade do evento, encontrou os destroços reluzentes e decidiu erguer ali sua morada, ligando em sete pontas uma muralha, abençoando a região com o nome Vallen.

Os fragmentos celestiais transformaram a floresta em um lugar mágico, gerando frutos e transformando-a em um paraíso na terra. Ao longo de séculos, os nômades formaram um vilarejo com a missão de proteger os cristais, garantindo a harmonia e a prosperidade da Floresta abençoada.

PROBLEMA

Otavius, um habitante do vilarejo, era um cientista fascinado pelos cristais, firmemente convicto de seu potencial extraordinário.

Sua mente inquieta e ambiciosa o levou a substituir um dos cristais por uma pedra semelhante para seu uso pessoal, sem que a aldeia percebesse. Ao aplicá-la em um colar, ele descobriu que conseguia aumentar sua força e vitalidade.

1ª ESCOLHA

Você, um jovem morador da aldeia, está caminhando e observa algo estranho acontecendo, decide:

"Ao criar um personagem nesse mundo, e jogar o jogo interagindo e desenvolvendo ao longo do jogo, as características do seu "avatar"

[1] Investigar para ver se tem algo rolando realmente.

[2] Deixar para lá e continuar a caminhar.

DESCOBERTA

Guiado pela intuição, você caminha até os cristais e descobre os mistérios: animais inquietos, e a flora murchando. Uma pedra estranha entre os cristais denuncia um roubo, enquanto anotações estranhas jogadas no chão denunciam a suspeita sobre Dr. Otavius.

DESCONFIANÇAS

Enquanto percorre a aldeia, sinais chamam sua atenção: a flora murchando, e animais agitados. Papeis espalhados pelo chão mencionam os cristais. Uma aura de mistério envolve você, aumentando sua suspeita sobre Otavius, o cientista local.

DIALOGO

Preocupado, você chama Sarah, sua melhor amiga para caminhar ao anoitecer, a fim de contar suas preocupações.

Você: Sarah! encontrei uma pedra estranha entre os cristais e anotações suspeitas no chão. Parece que alguém está mexendo com os cristais, talvez até roubando-os.

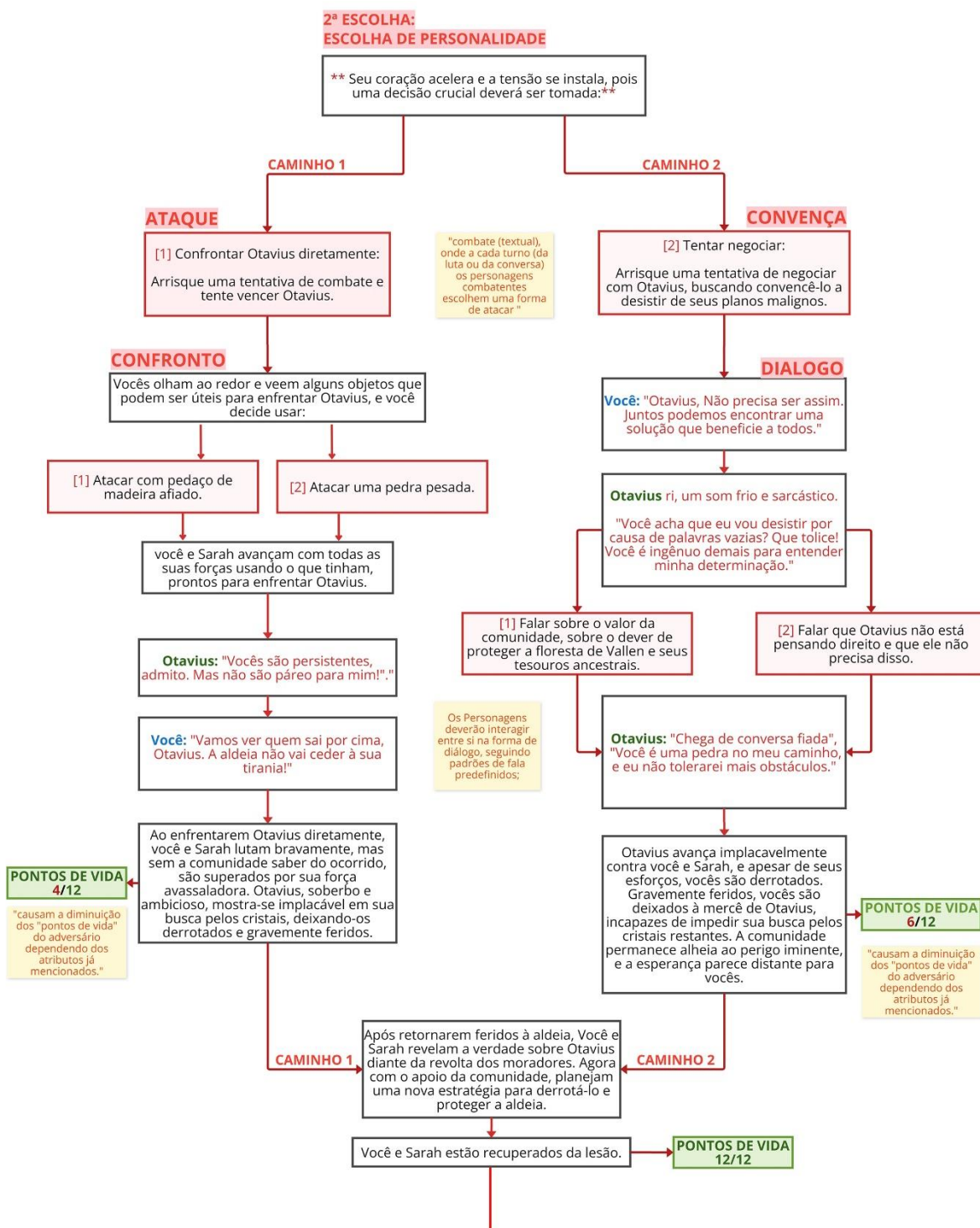
Sarah: Bem que eu estava suspeitando de algo na floresta, percebi que os animais estão aflitos e os frutos murcharam... as coisas parecem que estão perdendo a vida eu suspeito de alguém....

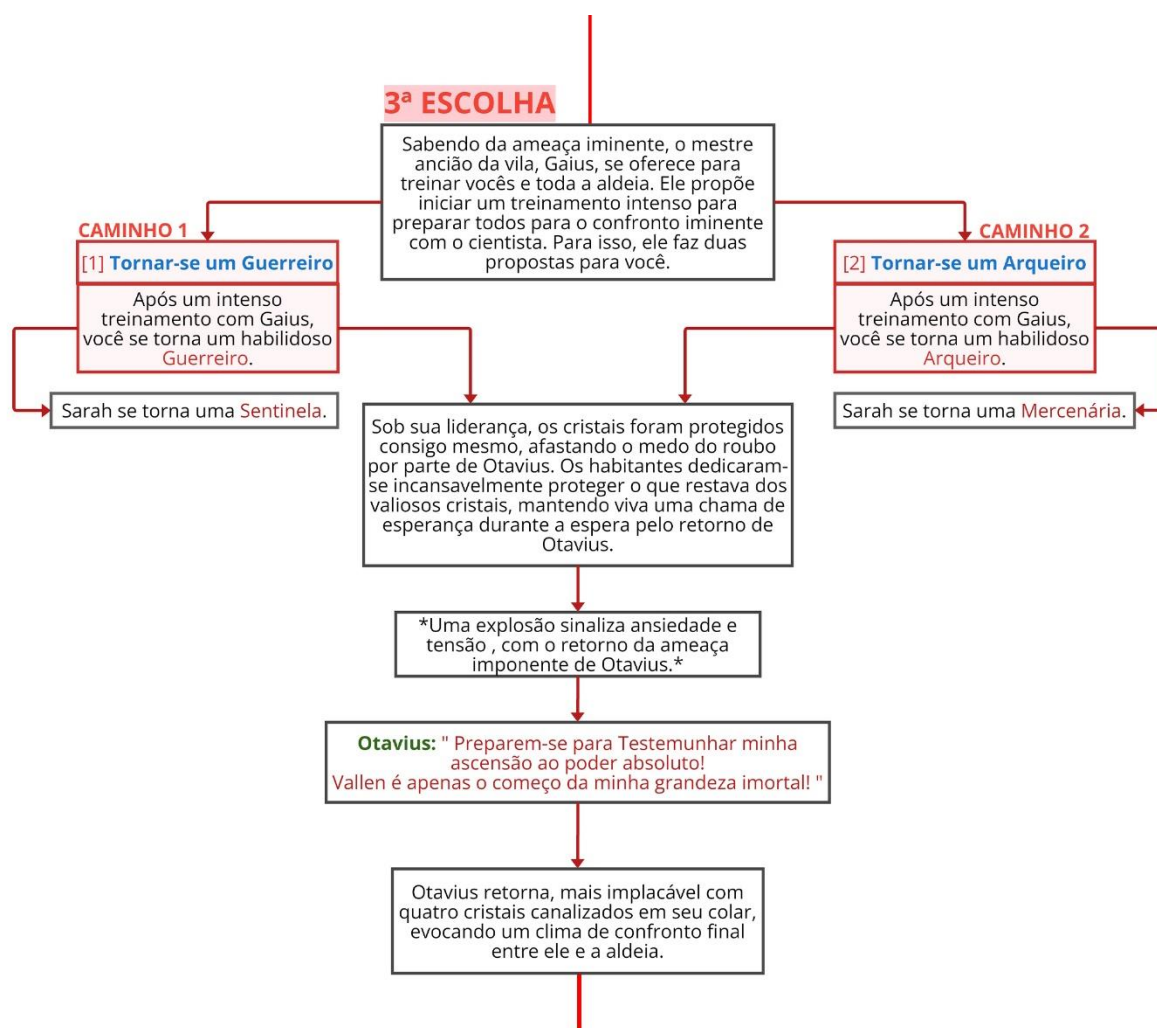
Você e Sarah: Dr. Otavius!!!!

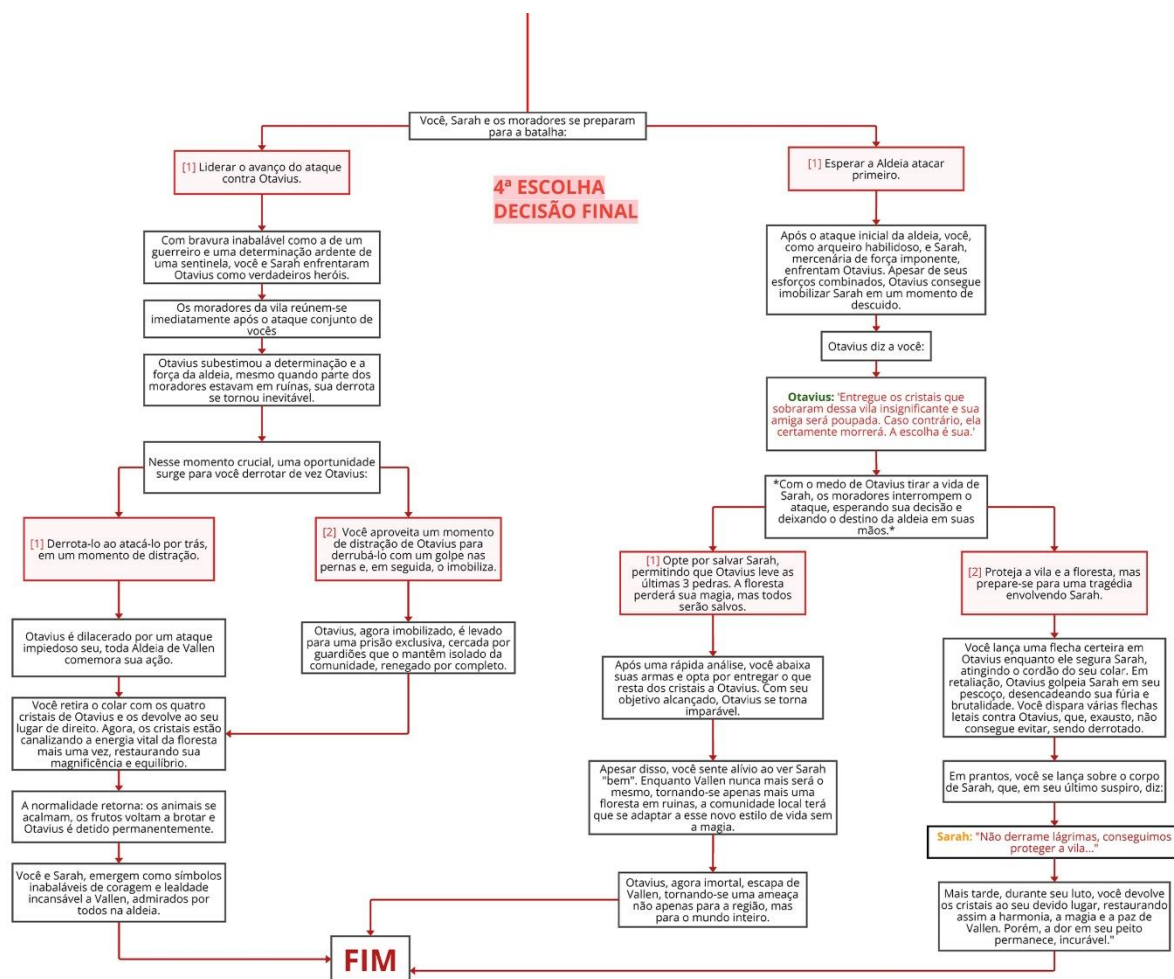
Caminhando, Você e Sarah, perto da muralha da aldeia, observam Otavius, indo em direção as muralhas adornadas com os cristais. Em seu pescoço, observa seu colar canalizado com três pedras. Você lembra do conteúdo das anotações perdidas, que fala:

"Canalizar as sete pedras garantiria imortalidade, mas cada fragmento aumentaria a força e vitalidade, embora causasse fadiga."

Otavius percebe a presença de ambos, e agora eles estão em perigo."







Durante o processo de codificação do jogo, as informações do fluxograma serviram como base, porém, algumas adaptações foram realizadas para otimizar o funcionamento do código e a experiência do jogador. Detalhes foram removidos ou editados visando melhorar a eficiência do jogo, garantindo que as funcionalidades sejam executadas de forma fluida e proporcionando uma experiência envolvente e satisfatória para o usuário.

REFERENCIAL TEÓRICO:

Nesse projeto foi utilizado referencias teóricas vistas em sala de aula e laboratório como:

Classe e Método main

```
//CLASSE ONDE O JOGO SERÁ RODADO
```

```
public class Principal {
```

```
// METODO MAIN
```

```
public static void main(String[] args) {
```

```
String msg, status;
```

O código apresenta uma classe chamada "Principal", que serve como ponto de entrada para o programa, onde o jogo será executado. O método "main" é o ponto de partida da execução do programa, sendo responsável por iniciar a aplicação. Dentro deste método, é declarada uma variável de tipo String chamada "msg" e outra chamada "status", que serão utilizadas para armazenar mensagens ou informações sobre o status das personagens durante sua execução.

Encapsulamento

```
//CLASSE PERSONAGEM
```

```
public class Personagem {
```

```
//CLASSE PERSONAGEM ENCAPSULADA
```

```
protected String nome;
```

```
protected int vida;
```

```
protected int forca;
```

```
protected int agilidade;
```

Este código define uma classe chamada "Personagem" que encapsula os atributos "nome", "vida", "força" e "agilidade". Esses atributos são protegidos, o que significa que podem ser acessados por subclasses da classe "Personagem", mas não por classes fora do pacote onde a classe "Personagem" está definida.

Métodos Construtores

//METODO CONSTRUTOR PERSONAGEM

```
public Personagem() {  
  
}
```

Este trecho de código define o construtor padrão da classe "Personagem". Em Java, um construtor é um bloco de código especial que é chamado quando uma nova instância de uma classe é criada. O construtor padrão fornecido aqui não possui parâmetros e não realiza nenhuma ação específica. Ele simplesmente permite a criação de objetos da classe "Personagem" com seus atributos iniciais não definidos ou com os valores padrão do Java (como null para objetos e 0 para tipos primitivos). Mesmo que este construtor não faça nada explicitamente, sua presença é importante para permitir a instanciação da classe "Personagem" sem a necessidade de passar argumentos.

Getters e Setters

//METODOS GETTERS E SETTERS

```
public String getNome() {  
  
    return nome;  
  
}  
  
public void setNome(String nome) {  
  
    this.nome = nome;
```

```
}

public int getVida() {
    return vida;
}

public void setVida(int vida) {
    this.vida = vida;
}

public int getForca() {
    return forca;
}

public void setForca(int forca) {
    this.forca = forca;
}

public int getAgilidade() {
    return agilidade;
}

public void setAgilidade(int agilidade) {
    this.agilidade = agilidade;
}
```

Este trecho de código define os métodos getters e setters para a classe "Personagem". Os getters (getNome, getVida, getForca, getAgilidade) retornam os valores dos atributos correspondentes. Os setters (setNome, setVida, setForca, setAgilidade) permitem definir novos valores para esses atributos. Esses métodos facilitam o acesso e a modificação dos atributos privados ou protegidos da classe, promovendo o encapsulamento e a integridade dos dados.

Criação de Objeto

```
// CRIANDO AS PERSONAGENS
```

```
    Personagem jogador = new Personagem();  
  
    jogador.setNome("Jogador");  
  
    jogador.setVida(22);  
  
    jogador.setForca(12);  
  
    jogador.setAgilidade(7);
```

Neste trecho de código, estão sendo criadas instâncias da classe "Personagem" para representar os personagens do jogo. A variável "jogador" é inicializada com uma nova instância da classe "Personagem", e em seguida, são definidos os atributos desse personagem através dos métodos "setNome", "setVida", "setForca" e "setAgilidade", atribuindo valores específicos para o nome, vida, força e agilidade do jogador. Esses métodos de "set" são responsáveis por definir os valores dos atributos do objeto "jogador", permitindo a configuração personalizada de cada personagem do jogo.

Scanner

```
import java.util.Scanner;
```

```
Scanner escolha = new Scanner(System.in);
```

```
    int escolhaUm = escolha.nextInt();
```

```
    if (escolhaUm == 1) {
```

```
        msg = "\r";
```

```
        msg += "Guiado pela intuição, você caminha até os cristais e desvenda os  
mistérios:\r";
```

```
        msg += "animais inquietos, e a flora murchando. Uma pedra estranha entre os  
cristais denuncia\r";
```

```
msg += "um roubo, enquanto anotações estranhas jogadas no chão denunciam a
suspeitas sobre Dr.Otavius\r";
```

```
System.out.println(msg);
```

```
} else if (escolhaUm == 2) {
```

```
msg = "\r";
```

```
msg += "Enquanto percorre a aldeia, sinais chamam sua atenção: a flora
murchando,\r";
```

```
msg += "e animais agitados. Papeis espalhados pelo chão mencionam os
cristais.\r";
```

```
msg += "Uma aura de mistério envolve você aumentando sua suspeitas sobre
Otavius, o cientista local.\r";
```

```
System.out.println(msg);
```

```
} else {
```

```
System.out.println("Tome um Decisão");
```

```
System.exit(0);
```

```
}
```

O código utiliza a classe Scanner para capturar a entrada do usuário a partir do console. O objeto escolha é criado como uma instância da classe Scanner que lê a entrada do sistema (System.in). A variável escolhaUm armazena o número inteiro inserido pelo usuário através do método nextInt(). Com base no valor de escolhaUm, diferentes mensagens são construídas e exibidas utilizando System.out.println(). Se o valor não for 1 ou 2, o programa imprime "Tome uma Decisão" e encerra a execução com System.exit(0).

If e Else

```
int e2c1 = escolha.nextInt(); // ESCOLHA 2 CAMINHO 1
```

```

        if (e2c1 == 1 || e2c1 == 2) {
            msg = "\r";
            msg += "Vocês avançam com todas as suas forças usando o que tinham,\r";
            msg += "prontos para enfrentar Otavius.\r";

            System.out.println(msg);
        } else {
            System.out.println("Tome uma Decisão");
            System.exit(0);
        }
    }

```

O código utiliza uma estrutura de controle if-else para direcionar o fluxo do programa com base na entrada do usuário, lida através da variável `e2c1` usando `Scanner.nextInt()`. Se o valor de `e2c1` for 1 ou 2, uma mensagem é construída e exibida, indicando que os personagens avançam para enfrentar Otavius. Caso contrário, uma mensagem solicitando uma decisão é exibida e o programa é encerrado com `System.exit(0)`. Este controle condicional garante que apenas entradas válidas permitam a progressão do jogo, mantendo a integridade do fluxo narrativo.

Herança

```

//CLASSE GUERREIRO QUE ESTÁ HERDANDO PERSONAGEM

public class Guerreiro extends Personagem {

    //METODO CONSTRUTOR GUERREIRO

    public Guerreiro(String nome, int vida, int forca, int agilidade) {

        this.nome = nome;

        this.vida = vida;

        this.forca = forca;
    }
}

```



```
        this.agilidade = agilidade;  
    }  
}
```

Neste código, a classe Guerreiro herda da classe Personagem, utilizando o conceito de herança da Programação Orientada a Objetos (OOP). Isso significa que Guerreiro é uma subclasse de Personagem e, portanto, herda todos os atributos e métodos da classe Personagem. O construtor da classe Guerreiro recebe parâmetros para inicializar os atributos nome, vida, força e agilidade, utilizando a palavra-chave `this` para referenciar os atributos herdados da classe pai Personagem. Isso permite criar instâncias de Guerreiro com atributos específicos definidos no momento da criação.

REGRAS E FUNCIONAMENTO DO JOGO:

O objetivo principal de "Vallen" é proteger os cristais sagrados da aldeia de Vallen e derrotar o malfeitor, o cientista ambicioso que ameaça a harmonia da floresta. O jogador deve trabalhar juntos com sua companheira para enfrentar desafios, fazer escolhas estratégicas e alcançar um dos três finais possíveis, determinados pelas decisões tomadas ao longo da jornada.

Criação de personagem:

A criação de personagem é um processo predefinido, no qual tanto o jogador quanto os personagens que compõem a trama possuem características e atributos estabelecidos previamente. Essa determinação prévia das características dos personagens contribui para a consistência e coesão da narrativa, permitindo que os jogadores se engajem em um mundo virtual coeso e imersivo. A construção do personagem segue um modelo fixo, no qual as características e habilidades dos personagens são delineadas de acordo com as necessidades da trama e as dinâmicas estabelecidas pelo jogo. Dessa forma, a criação de personagem é um elemento fundamental no desenvolvimento da experiência de jogo, fornecendo ao jogador um ponto de partida claro e definido para sua jornada no universo virtual apresentado pelo jogo.

Componentes do Jogo:

Os componentes do jogo "Vallen" compõem-se de elementos essenciais que contribuem para a construção e imersão no universo virtual apresentado aos jogadores. A representação deste universo é materializada através da aldeia de Vallen

e da floresta mágica que a circunda, constituindo o cenário principal onde a narrativa se desenrola. Dentro deste contexto, os jogadores são apresentados a uma variedade de escolhas de eventos e conflitos, os quais são fundamentais para o desenvolvimento da trama e para a progressão do jogo.

Além disso, o jogo dispõe de escolhas, encontros narrativos, eventos, que desempenham um papel crucial na impulsão da narrativa, ao mesmo tempo em que apresentam desafios aos jogadores. Estas escolhas fornecem ao jogador uma gama diversificada de situações a serem enfrentados, adicionando complexidade e imprevisibilidade à experiência de jogo.

Por fim, o componente de resolução de conflitos e determinação de resultados predefinidos é alcançado por meio de mecânicas como combate e eventos. Estas mecânicas fornecem aos jogadores as ferramentas necessárias para enfrentar desafios, resolver conflitos e avançar na narrativa, contribuindo para a imersão e a interatividade do jogo.

No conjunto, esses componentes formam as principais ferramentas à disposição do jogador para explorar o mundo de Vallen, interagir com os desafios apresentados e avançar na narrativa emocionante do jogo, proporcionando uma experiência envolvente e repleta de possibilidades.

Ações do Jogador:

O jogador desempenha um papel crucial na condução da narrativa e no avanço do jogo, oferecendo uma variedade de opções para interagir com o ambiente virtual e influenciar o desenrolar da história. Uma dessas ações fundamentais é a interpretação, na qual os jogadores têm a oportunidade de interagir com personagens, diálogos predefinidos, encontros e eventos apresentados no jogo. Ao fazer escolhas estratégicas e decisões significativas durante essas interações, os jogadores podem moldar o curso da narrativa e determinar o resultado de eventos futuros. Essa

capacidade de interpretação não apenas permite aos jogadores uma imersão mais profunda no mundo virtual do jogo, mas também promove um engajamento ativo com a trama e uma sensação de agência sobre o desenvolvimento da história.

Cooperação:

A cooperação é um elemento fundamental no jogo "Vallen", onde o jogador se une para resolver problemas e enfrentar desafios ao lado de seus aliados. Ao combinar suas habilidades individuais e recursos, eles trabalham em conjunto para alcançar objetivos comuns dentro do jogo. Essa colaboração entre o jogador e seus aliados permite uma exploração estratégica e imersiva do mundo de Vallen, onde enfrentam desafios diversos e tomam decisões que têm um impacto direto no desenvolvimento da história. Essa dinâmica de cooperação não só fortalece a experiência de jogo, mas também promove um senso de companheirismo e trabalho em equipe entre os participantes, reforçando a interação e a imersão no universo do jogo.

Resolução de Conflitos:

A resolução de conflitos no jogo "Vallen" é uma combinação de elementos narrativos e estratégicos, os quais refletem as escolhas e habilidades dos jogadores. No âmago dessa resolução está o combate textual, uma mecânica na qual os jogadores se deparam com desafios apresentados de forma textual e são confrontados com a necessidade de fazer escolhas estratégicas para superar seus oponentes. Este método de resolução de conflitos é cuidadosamente projetado para proporcionar uma experiência envolvente e desafiadora, incentivando os jogadores a empregar suas melhores habilidades e estratégias para alcançar o objetivo pretendido. Ao enfrentar os conflitos de forma narrativa e estratégica, "Vallen" promove uma imersão profunda no mundo do jogo, onde as escolhas dos jogadores têm um impacto significativo no desenrolar da história e no resultado final do jogo.

Condições de Vitória:

A conquista da vitória no jogo requer do jogador a seleção de um percurso incerto, pois não há uma trajetória predefinida para o triunfo, sendo possível encaminhar-se para três desfechos distintos. Estes desfechos delineiam-se como: uma rota triunfante, onde o jogador alcança a vitória sem grandes obstáculos; uma rota na qual a vitória é possível, porém mediante um sacrifício impactante que reverbera na narrativa; e por fim, uma rota que conduz à ruína, marcada por consequências desfavoráveis e a derrota do jogador. Essa incerteza inerente ao desfecho do jogo adiciona complexidade e imprevisibilidade à experiência do jogador, desafiando-o a fazer escolhas ponderadas e estratégicas ao longo do percurso narrativo.

Regras de Funcionamento:

No contexto do jogo, é fundamental que o jogador faça uma das escolhas predefinidas pelo sistema para garantir o bom funcionamento da experiência de jogo. Essas escolhas predefinidas são elementos-chave que impulsionam a narrativa e influenciam o desenvolvimento da trama. A falta de uma escolha por parte do jogador resultaria em um impasse no progresso do jogo, impedindo a continuidade da narrativa e comprometendo a experiência do usuário.

A obrigatoriedade de fazer uma escolha predefinida pelo jogo é estabelecida para promover a participação ativa do jogador na trama e garantir sua interação com o ambiente virtual. Ao tomar uma decisão, o jogador contribui para a progressão da história e molda o desenrolar dos eventos. No entanto, a não realização de uma escolha resultaria em um estado de estagnação no jogo, requerendo que o jogador reinicie a partida e retorne ao ponto inicial para tentar novamente.

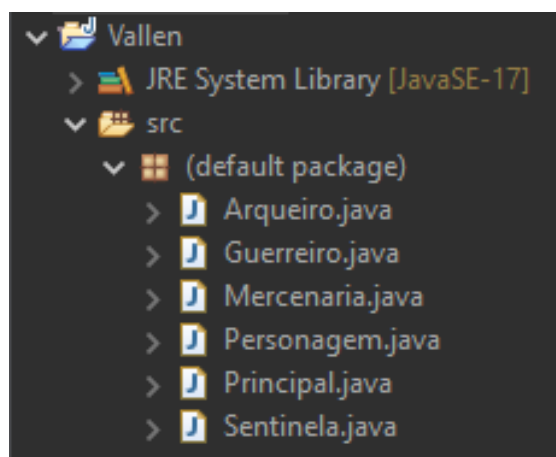
Portanto, a obrigatoriedade de fazer uma escolha predefinida pelo jogo é um aspecto crucial para o funcionamento adequado da experiência de jogo, garantindo uma narrativa coesa e envolvente que responde às ações e decisões do jogador.

ESTRUTURA DO PROGRAMA:

O projeto "Vallen" é uma aplicação desenvolvida em Java que simula uma história na qual você faz suas escolhas de combate entre diferentes tipos de cenário. Cada personagem possui atributos e comportamentos específicos, refletindo a diversidade das classes de combate típicas em jogos de RPG (Role Playing Games). Este documento apresenta uma visão detalhada da estrutura do projeto, explicando a organização dos arquivos e a funcionalidade de cada componente.

Estrutura de Diretórios e Arquivos

A organização do projeto "Vallen" segue uma estrutura simplificada, onde todas as classes estão no pacote padrão. A estrutura é a seguinte:



JRE System Library [JavaSE-17]

Este item refere-se à biblioteca de tempo de execução do Java SE 17, necessária para a compilação e execução do projeto. Ela fornece todas as classes e interfaces padrão da linguagem Java.

No projeto "Vallen", as classes "Arqueiro", "Guerreiro", "Mercenária" e "Sentinela" foram codificadas de maneira iguais com o propósito de permitir que futuras personagens herdem essas classes. Essa abordagem foi adotada visando promover a reutilização de código e facilitar a extensibilidade do sistema. Ao manter uma

estrutura comum entre essas classes, torna-se mais simples adicionar novos tipos de personagens ao projeto, mantendo a consistência e a coesão do código. Isso também promove uma manutenção mais eficiente, já que as alterações feitas em uma classe base refletirão automaticamente em todas as suas subclasses.

```
//CLASSE ARQUEIRO QUE ESTÁ HERDANDO PERSONAGEM
```

```
public class Arqueiro extends Personagem {
```

```
//METODO CONSTRUTOR ARQUEIRO
```

```
    public Arqueiro(String nome, int vida, int forca, int agilidade) {
```

```
        this.nome = nome;
```

```
        this.vida = vida;
```

```
        this.forca = forca;
```

```
        this.agilidade = agilidade;
```

```
    }
```

```
}
```

Na classe Personagem, os atributos-base das personagens foram definidos e encapsulados para facilitar a criação de objetos em outras classes usando os métodos Getters e Setters. Esse encapsulamento permite o controle do acesso aos atributos, garantindo a integridade dos dados e facilitando a manutenção do código. Os métodos Getters permitem acessar os valores dos atributos de forma segura, enquanto os Setters possibilitam a modificação desses valores conforme as regras de negócio estabelecidas na classe Personagem.

```
//CLASSE PERSONAGEM
```

```
public class Personagem {
```

```
//CLASSE PERSONAGEM ENCAPSULADA
```

```
protected String nome;

protected int vida;

protected int forca;

protected int agilidade;

//METODO CONSTRUTOR PERSONAGEM

public Personagem() {

}

//METODOS GETTERS E SETTERS

public String getNome() {

    return nome;

}

public void setNome(String nome) {

    this.nome = nome;

}

public int getVida() {

    return vida;

}

public void setVida(int vida) {

    this.vida = vida;

}

public int getForca() {

    return forca;

}

public void setForca(int forca) {
```



```

        this.forca = forca;
    }

    public int getAgilidade() {
        return agilidade;
    }

    public void setAgilidade(int agilidade) {
        this.agilidade = agilidade;
    }
}

```

Na classe Principal, encontra-se o método main, responsável por iniciar a execução do programa. Além disso, nela é desenvolvido o enredo do jogo, incluindo a criação das personagens e as lógicas de combate. Essa classe desempenha um papel central na execução do programa, coordenando as interações entre os elementos do jogo e garantindo o seu funcionamento adequado.

```

import java.util.Scanner;

//CLASSE ONDE O JOGO SERÁ RODADO

public class Principal {

    // METODO MAIN

    public static void main(String[] args) {

        String msg, status;

        msg = "Floresta VALLEN:\n";
    }
}

```

Maiores detalhes sobre o código principal serão abordados no próximo tópico.

RELATÓRIO DO CÓDIGO:

```
import java.util.Scanner;

//CLASSE ONDE O JOGO SERÁ RODADO

public class Principal {

    // METODO MAIN

    public static void main(String[] args) {

        String msg, status;

        msg = "Floresta VALLEN:\n";

        msg += "\r";

        msg += "\r";

        msg += "No princípio, um cometa caiu na terra, transformando uma antiga floresta\r";

        msg += "em seu santuário e espalhando seus cristais mágicos em sete direções do lugar.\r";

        msg += "Diante desse espetáculo celestial, uma tribo de nômades, tocada pela grandiosidade\r";

        msg += "do evento, encontrou os destroços reluzentes e decidiu erguer ali sua morada,\r";

        msg += "ligando em sete pontas uma muralha, abençoando a região com o nome Vallen.\r";

        msg += "Os fragmentos celestiais transformaram a floresta em um lugar mágico,\r";

        msg += "gerando frutos e transformando-a em um paraíso na terra.\r";

        msg += "Ao longo de séculos os nômades formaram um vilarejo com a missão de proteger os\r";

        msg += "cristais, garantindo a harmonia e a prosperidade da Floresta abençoada.\r";

        System.out.println(msg);

        msg = "Otavius, um habitante do vilarejo, era um cientista fascinado\r";
```

```
msg += "pelos cristais, firmemente convicto de seu potencial extraordinário\r";  
msg += "Sua mente inquieta e ambiciosa o levou a substituir um dos cristais por\r";  
msg += "uma pedra semelhante para seu uso pessoal, sem que a aldeia  
percebesse\r";  
msg += "Ao aplicá-la em um colar, ele descobriu que conseguia aumentar\r";  
msg += "sua força e vitalidade.\r";  
  
    System.out.println(msg);  
  
msg = "Você, um jovem morador da aldeia, está caminhando\r";  
msg += "e observa algo estranho acontecendo, decide\r";  
msg += "\r";  
msg += "[1] Investigar para ver se tem algo rolando realmente\r";  
msg += "[2] Deixar para lá e continuar a caminhar\r";  
  
    System.out.println(msg);  
  
    // PRIMEIRA ESCOLHA  
  
Scanner escolha = new Scanner(System.in);  
  
    int escolhaUm = escolha.nextInt();  
  
    if (escolhaUm == 1) {  
  
msg = "\r";  
  
msg += "Guiado pela intuição, você caminha até os cristais e desvenda os  
mistérios:\r";  
  
msg += "animais inquietos, e a flora murchando. Uma pedra estranha entre os  
cristais denuncia\r";  
  
msg += "um roubo, enquanto anotações estranhas jogadas no chão denunciam a  
suspeitas sobre Dr.Otavius\r";  
  
        System.out.println(msg);  
  
    } else if (escolhaUm == 2) {
```

```
msg = "\r";

msg += "Enquanto percorre a aldeia, sinais chamam sua atenção: a flora
murchando,\r";

msg += "e animais agitados. Papeis espalhados pelo chão mencionam os cristais.\r";

msg += "Uma aura de mistério envolve você aumentando sua suspeitas sobre
Otavius, o cientista local.\r";

        System.out.println(msg);

    } else {

        System.out.println("Tome uma Decisão");

        System.exit(0);

    }

    // FIM PRIMEIRA ESCOLHA

msg = "Preocupado, você chama Sarah, sua melhor amiga para caminhar\r";

msg += "ao anoitecer, a fim de contar suas preocupações.\r";

msg += "\r";

msg += "Jogador: Sarah! encontrei uma pedra estranha entre os cristais e anotações
suspeitas no chão.\r";

msg += "Parece que alguém está mexendo com os cristais, talvez até roubando-
os.\r";

msg += "\r";

msg += "Sarah: Bem que eu estava suspeitando de algo na floresta, percebi que
os\r";

msg += "animais estão aflitos e os frutos murcharam... \r";

msg += "as coisas parecem que estão perdendo a vida! eu suspeito de alguém...\r";

msg += "\r";

msg += "Jogador e Sarah: Dr. Otavius!!!!\r";
```

```

msg += "\r";

msg += "Caminhando, Você e Sarah, perto da muralha da aldeia, observam Otavius,
indo\r";

msg += "em direção as muralhas adornada com os cristais. Em seu pescoço,
observa seu\r";

msg += "colar canalizado com três pedras. Você lembra do conteúdo das anotações
perdidas, que fala:\r";

msg += "\r";

msg += "Canalizar as sete pedras garantiria imortalidade, mas cada fragmento\r";

msg += "aumentaria a força e vitalidade, embora causasse fadiga.\r";

// CADA PEDRA AUMENTA 8 DE VIDA E 5 DE FORÇA E DIMINUI 1 DE
VELOCIDADE

msg += "Otavius percebe a presença de ambos, e agora eles estão em perigo.\r";

msg += "\r";

msg += "Seu coração acelera e a tensão se instala, pois uma decisão crucial\r";

msg += "deverá ser tomada:\r";

        System.out.println(msg);


// CRIANDO AS PERSONAGENS

        Personagem jogador = new Personagem();

        jogador.setNome("Jogador");

        jogador.setVida(22);

        jogador.setForca(12);

        jogador.setAgilidade(7);

        Personagem sarah = new Personagem();

        sarah.setNome("Sarah");

```

```
sarah.setVida(22);

sarah.setForca(12);

sarah.setAgilidade(7);

Personagem otavius = new Personagem();

otavius.setNome("Otavius");

otavius.setVida(46);

otavius.setForca(27);

otavius.setAgilidade(4);
```

// MOSTANDO OS STATUS DOS PERSONAGENS

```
status = "-----\r";

status += jogador.getNome() + "\r";

status += "Vida: " + jogador.getVida() + "\r";

status += "Força: " + jogador.getForca() + "\r";

status += "Agilidade: " + jogador.getAgilidade() + "\r";

status += "-----\r";

status += sarah.getNome() + "\r";

status += "Vida: " + sarah.getVida() + "\r";

status += "Força: " + sarah.getForca() + "\r";

status += "Agilidade: " + sarah.getAgilidade() + "\r";

status += "-----\r";

status += otavius.getNome() + "\r";

status += "Vida: " + otavius.getVida() + "\r";

status += "Força: " + otavius.getForca() + "\r";

status += "Agilidade: " + otavius.getAgilidade() + "\r";
```

```

        status += "-----\r";

        System.out.println(status);

// SEGUNDA ESCOLHA

msg = "[1] Confrontar Otavius diretamente:\r";

msg += "Arrisque uma tentativa de combate e tente vencer Otavius.\r";

msg += "[2] Tentar negociar:\r";

msg += "Arrisque uma tentativa de negociar com Otavius, buscando convencê-lo a
desistir de seus planos malignos.\r";

        System.out.println(msg);

        int escolhaDois = escolha.nextInt();

        if (escolhaDois == 1) {

msg = "\r";

msg += "Vocês olham ao redor e veem alguns objetos que\r";

msg += "podem ser úteis para enfrentar Otavius, e você\r";

msg += "decide usar:\r";

msg += "\r";

msg += "[1] Atacar com pedaço de madeira afiado.\r";

msg += "[2] Atacar com uma pedra pesada.\r";

                System.out.println(msg);

                int e2c1 = escolha.nextInt(); // ESCOLHA 2 CAMINHO 1

                if (e2c1 == 1 || e2c1 == 2) {

msg = "\r";

msg += "Vocês avançam com todas as suas forças usando o que tinham,\r";

msg += "prontos para enfrentar Otavius.\r";

                        System.out.println(msg);

```

```

        } else {

            System.out.println("Tome um Decisão");

            System.exit(0);

        }

msg = "Otavius: Vocês são persistentes, admito. Mas não são páreo para mim!\r";

msg += "\r";

msg += "Você: Vamos ver quem sai por cima, Otavius. A aldeia não vai ceder à sua
tirania!\r";

msg += "\r";

msg += "Ao enfrentarem Otavius diretamente, você e Sarah lutam bravamente, mas
sem\r";

msg += "a comunidade saber do ocorrido, são superados por sua força
avassaladora. \r";

msg += "Otavius, soberbo e ambicioso, mostra-se implacável em sua busca pelos
cristais,\r";

msg += "deixando-os derrotados e gravemente feridos.\r ";

        System.out.println(msg);

    } else if (escolhaDois == 2) {

msg = "\r";

msg += "Você: Otavius, Não precisa ser assim. Juntos podemos encontrar uma
solução\r";

msg += "que beneficie a todos.\r";

msg += "\r";

msg += "Otavius ri, um som frio e sarcástico.\r";

msg += "\r";

msg += "Otavius: Você acha que eu vou desistir por causa de palavras vazias?\r";

```



```

msg += "Que tolice! Você é ingênuo demais para entender minha determinação.\r";

msg += "\r";

msg += "[1] Falar sobre o valor da comunidade e o dever de proteger\r";

msg += "a floresta de Vallen e seus tesouros ancestrais.\r";

msg += "[2] Falar que ele não está pensando direito e que ele não precisa disso.\r";

        System.out.println(msg);

        int e2c2 = escolha.nextInt(); // ESCOLHA 2 CAMINHO 2

        if (e2c2 == 1 || e2c2 == 2) {

msg = "\r";

msg += "Otavius:Chega de conversa fiada!\r";

msg += "Você é uma pedra no meu caminho, e eu não tolerarei mais obstáculos.\r";

                System.out.println(msg);

        } else {

                System.out.println("Tome uma Decisão");

                System.exit(0);

        }

msg = "Otavius avança implacavelmente contra você e Sarah, e apesar de seus
esforços,\r";

msg += "você são derrotados. Gravemente feridos, você são deixados à mercê de
Otavius,\r";

msg += "incapazes de impedir sua busca pelos cristais restantes.\r";

msg += "A comunidade permanece alheia ao perigo iminente, e a esperança parece
distante para você.\r";

        System.out.println(msg);

    } else {

```

```

        System.out.println("Tome um Decisão");

        System.exit(0);

    }

// FIM SEGUNDA ESCOLHA

// LOGICA DE BATALHA STATUS JOGADOR + SARAH > OTAVIUS

    int sv = jogador.getVida() + sarah.getVida(); // SOMA VIDA JOGADOR
+ SARAH

    int sf = jogador.getForca() + sarah.getForca(); // SOMA FORÇA
JOGADOR + SARAH

    status = "-----\r";

    status += jogador.getNome() + " + " + sarah.getNome() + "\r";

    status += "Vida: " + sv + "\r";

    status += "Força: " + sf + "\r";

    status += "-----\r";

    status += otavius.getNome() + "\r";

    status += "Vida: " + otavius.getVida() + "\r";

    status += "Força: " + otavius.getForca() + "\r";

    status += "-----\r";

    System.out.println(status);

    if (sv > otavius.getVida() && sf > otavius.getForca()) {

        msg = "Você e sarah derrotam Otavius";

        System.out.println(msg);

    } else {

        msg = "Você e Sarah ainda nao supera a força de Otavius, ";

        msg += "e foram derrotados";

```

```

        System.out.println(msg);

    }

    msg = "Após retornarem feridos à aldeia,\r";

    msg += "Você e Sarah revelam a verdade sobre Otavius, diante da revolta dos moradores.\r";

    msg += "Agora, com o apoio da comunidade,\r";

    msg += "planejam uma nova estratégia para derrotá-lo e proteger a aldeia.\r";

    msg += "\r";

    msg += "Você e Sarah estão recuperados da lesão.\r";

    msg += "\r";

    msg += "Sabendo da ameaça iminente, o mestre ancião da vila,\r";

    msg += "Gaius, se oferece para treinar você e Sarah.\r";

    msg += "Ele propõe iniciar um treinamento intenso para\r";

    msg += "preparar todos para o confronto iminente com o cientista.\r";

    msg += "Para isso, ele faz duas propostas para você!\r";

    msg += "\r";

    msg += "[1] Você se tornara um guerreiro e Sarah uma sentinela\r";

    msg += "[2] Você se tornara um arqueiro e Sarah uma Mercenaria\r";

        System.out.println(msg);

// TERCEIRA ESCOLHA

        int escolhaTres = escolha.nextInt();

        if (escolhaTres == 1) {

            int jogadorVida, jogadorForca, sarahVida, sarahForca, sVida,
sForca, otaviusVida, otaviusForca;

```

```

    Guerreiro guerreiroJogador = new Guerreiro("Jogador", 12, 10,
4);

    Sentinela sentinelaSarah = new Sentinela("Sarah", 8, 6, 2);

    jogadorVida = jogador.getVida() + guerreiroJogador.getVida();

    jogadorForca = jogador.getForca() +
guerreiroJogador.getForca();

    sarahVida = sarah.getVida() + sentinelaSarah.getVida();

    sarahForca = sarah.getForca() + sentinelaSarah.getForca();

msg = "\r";

msg += "Após um intenso treinamento com Gaius,\r";

msg += "você se torna um habilidoso Guerreiro\r";

msg += "e Sarah se torna uma Sentinela\r";

    System.out.println(msg);

    sVida = jogadorVida + sarahVida; // SOMA VIDA JOGADOR +
SARAH

    sForca = jogadorForca + sarahForca; // SOMA FORÇA
JOGADOR + SARAH

    otaviusVida = otavius.getVida() + 16; //NESSE MOMENTO
OTAVIUS ESTÁ COM DUAS PEDRAS A MAIS

    otaviusForca = otavius.getForca() + 10; //NESSE MOMENTO
OTAVIUS ESTÁ COM DUAS PEDRAS A MAIS

    status = "-----\r";

    status += jogador.getNome() + " + " + sarah.getNome() + "\r";

    status += "Vida: " + sVida + "\r";

    status += "Força: " + sForca + "\r";

    status += "-----\r";

```

```

        status += otavius.getNome() + "\r";

        status += "Vida: " + otaviusVida + "\r";

        status += "Força: " + otaviusForca + "\r";

        status += "-----\r";

        System.out.println(status);

    } else if (escolhaTres == 2) {

        int jogadorVida, jogadorForca, sarahVida, sarahForca, sVida,
sForca, otaviusVida, otaviusForca;

        Arqueiro arqueiroJogador = new Arqueiro("Jogador", 8, 4, 2);

        Mercenaria mercenariaSarah = new Mercenaria("Sarah", 10, 6,
7);

        jogadorVida = jogador.getVida() + arqueiroJogador.getVida();

        jogadorForca = jogador.getForca() + arqueiroJogador.getForca();

        sarahVida = sarah.getVida() + mercenariaSarah.getVida();

        sarahForca = sarah.getForca() + mercenariaSarah.getForca();

        msg = "\r";

        msg += "Após um intenso treinamento com Gaius,\r";

        msg += "você se torna um habilidoso Arqueiro\r";

        msg += "e Sarah se torna uma Mercenaria\r";

        System.out.println(msg);

        sVida = jogadorVida + sarahVida; // SOMA VIDA JOGADOR +
SARAH

        sForca = jogadorForca + sarahForca; // SOMA FORÇA
JOGADOR + SARAH

```

```

        otaviusVida = otavius.getVida() + 16; //NESSE MOMENTO
OTAVIUS ESTÁ COM DUAS PEDRAS A MAIS

```

```

        otaviusForca = otavius.getForca() + 10; //NESSE MOMENTO
OTAVIUS ESTÁ COM DUAS PEDRAS A MAIS

```

```

        status = "-----\r";

        status += jogador.getNome() + " + " + sarah.getNome() + "\r";

        status += "Vida: " + sVida + "\r";

        status += "Força: " + sForca + "\r";

        status += "-----\r";

        status += otavius.getNome() + "\r";

        status += "Vida: " + otaviusVida + "\r";

        status += "Força: " + otaviusForca + "\r";

        status += "-----\r";

        System.out.println(status);

    } else {

        System.out.println("Tome um Decisão");

        System.exit(0);

    }

```

```

//FIM TERCEIRA ESCOLHA

```

```

msg = "Sob sua liderança, os cristais foram protegidos consigo mesmo, \r";

msg += "afastando o medo do roubo por parte de Otavius. \r";

msg += "Os habitantes dedicaram-se incansavelmente proteger o que restava dos \r";

msg += "valiosos cristais, mantendo viva uma chama de \r";

msg += "esperança durante a espera pelo retorno de Otavius. \r";

```

```
System.out.println(msg);

msg = "Uma explosão sinaliza ansiedade e tensão,\r";

msg += "com o retorno da ameaça imponente de Otavius.\r";

System.out.println(msg);

msg = "Otavius: Preparem-se para Testemunhar minha ascensão ao poder absoluto!\r";

msg += "Vallen é apenas o começo da minha grandeza imortal!\r";

System.out.println(msg);

msg = "Otavius retorna, mais implacável com quatro cristais canalizados em seu colar, \r";

msg += "evocando um clima de confronto final entre ele e a aldeia. \r";

System.out.println(msg);

msg = "Você, Sarah e os moradores se preparam para a batalha:\r";

System.out.println(msg);

//CONSEQUENCIAS DA TERCEIRA ESCOLHA

//SE A TERCEIRA ESCOLHA FOI DE JOGADOR GUERREIRO E SARAH SENTINELA:

if (escolhaTres == 1) {

msg = "Com bravura inabalável como a de um guerreiro e uma determinação ardente de uma\r";

msg += "sentinela, você e Sarah enfrentaram Otavius como verdadeiros heróis.\r";

System.out.println(msg);

msg = "Os moradores da vila reúnem-se \r";

msg += "imediatamente após o ataque conjunto de vocês.\r";

System.out.println(msg);
```

```

msg = "Otavius subestimou a determinação e a força da aldeia, mesmo quando\r";
msg += "parte dos moradores estavam em ruínas, sua derrota é inevitável.\r";

    System.out.println(msg);

msg = "Nesse momento crucial, uma oportunidade\r";
msg += "surge para você derrotar de vez Otavius:\r";

    System.out.println(msg);

msg = "\r";
msg += "[1] Derrota-lo ao atacá-lo por trás, em um momento de distração.\r";
msg += "[2] Você aproveita um momento de distração de Otavius para derrubá-lo\r";
msg += "com um golpe nas pernas e, em seguida, o imobiliza.\r";

    System.out.println(msg);

    int e3c1 = escolha.nextInt(); // ESCOLHA TRES CAMINHO UM

    if (e3c1 == 1) {

msg = "\r";
msg += "Otavius é dilacerado por seu ataque impiedoso,\r";
msg += "toda Aldeia de Vallen comemora sua ação.\r";

        System.out.println(msg);

    } else if (e3c1 == 2) {

msg = "\r";
msg += "Otavius, agora imobilizado, é levado para uma prisão exclusiva,\r";
msg += "cercada por guardiões que o mantêm isolado da comunidade, renegado por completo.\r";

        System.out.println(msg);

    } else {

        System.out.println("Tome um Decisão");

```



```

        System.exit(0);

    }

    msg = "Você retira o colar com os quatro cristais de Otavius e os devolve ao seu
    lugar de direito.\r";

    msg += "Agora, os cristais estão canalizando a energia vital da floresta mais uma
    vez,\r";

    msg += "restaurando sua magnificência e equilíbrio.\r";

    System.out.println(msg);

    msg = "A normalidade retorna: os animais se acalmam,\r";

    msg += "os frutos voltam a brotar e Otavius é detido permanentemente.\r";

    System.out.println(msg);

    msg = "Você e Sarah, emergem como símbolos inabaláveis de coragem e
    lealdade\r";

    msg += "incansável a Vallen, admirados por todos na aldeia.\r";

    System.out.println(msg);

//SE A TERCEIRA ESCOLHA FOI DE JOGADOR ARQUEIRO E SARAH
MERCENARIA:

    } else if (escolhaTres == 2) {

    msg = "Após o ataque inicial da aldeia, você, como arqueiro habilidoso, e Sarah,
    mercenária\r";

    msg += "de força imponente, enfrentam Otavius. Apesar de seus esforços
    combinados,\r";

    msg += "Otavius consegue imobilizar Sarah em um momento de descuido. \r";

    System.out.println(msg);

    msg = "Otavius: Entregue os cristais que sobraram dessa vila insignificante e sua\r";

```

```
msg += "amiga será poupada. Caso contrário, ela certamente morrerá. A escolha é sua.\r";
```

```
System.out.println(msg);
```

```
msg = "Com o medo de Otavius tirar a vida de Sarah, os moradores interrompem o ataque,\r";
```

```
msg += "esperando sua decisão e deixando o destino da aldeia em suas mãos.\r";
```

```
System.out.println(msg);
```

```
msg = "[1] Opte por salvar Sarah, permitindo que Otavius leve as últimas três pedras.\r";
```

```
msg += "A floresta perderá sua magia, mas todos serão salvos.\r";
```

```
msg += "[2] Proteja a vila e a floresta, mas prepare-se para uma tragédia envolvendo Sarah.\r";
```

```
System.out.println(msg);
```

```
int e3c2 = escolha.nextInt(); // ESCOLHA TRES CAMINHO DOIS
```

```
if (e3c2 == 1) {
```

```
msg = "\r";
```

```
msg += "Após uma rápida análise, você abaixa suas armas e opta por entregar o que resta dos \r";
```

```
msg += "cristais a Otavius. Com seu objetivo alcançado, Otavius se torna imparável.\r";
```

```
System.out.println(msg);
```

```
msg = "Apesar disso, você sente alívio ao ver Sarah bem.\r";
```

```
msg += "Enquanto Vallen nunca mais será o mesmo, tornando-se apenas mais uma floresta \r";
```

```
msg += "em ruínas, a comunidade local terá que se adaptar a esse novo estilo de vida sem a magia.\r";
```

```

        System.out.println(msg);

msg = "Otavius, agora imortal, escapa de Vallen, tornando-se uma ameaça\r";
msg += "não apenas para a região, mas para o mundo inteiro.\r";

        System.out.println(msg);

    } else if (e3c2 == 2) {

msg = "\r";

msg += "Você lança uma flecha certa em Otavius enquanto ele segura Sarah,
atingindo o cordão do seu colar. \r";

msg += "Em retaliação, Otavius golpeia Sarah em seu pescoço, desencadeando sua
fúria e brutalidade. \r";

msg += "Você dispara várias flechas letais contra Otavius, que, exausto, não
consegue evitar, sendo derrotado. \r";

        System.out.println(msg);

msg = "Em prantos, você se lança sobre o corpo de Sarah, que, em seu último
suspiro, diz: \r";

        System.out.println(msg);

msg += "Sarah: Não derrame lágrimas, conseguimos proteger a vila... \r";

        System.out.println(msg);

msg = "Mais tarde, durante seu luto, você devolve os cristais ao seu devido lugar,
restaurando\r";

msg += "assim a harmonia, a magia e a paz de Vallen. Porém, a dor em seu peito
permanece, incurável.\r";

        System.out.println(msg);

    } else {

        System.out.println("Tome uma Decisão");

        System.exit(0);

```

```
        }  
    } else {  
        System.out.println("Tome um Decisão");  
        System.exit(0);  
    }  
    escolha.close();  
}  
  
}
```

// As classes que compõe o código principal não iriam caber só em 15 páginas

PROGRAMA EM FUNCIONAMENTO:

Floresta VALLEN:

No princípio, um cometa caiu na terra, transformando uma antiga floresta em seu santuário e espalhando seus cristais mágicos em sete direções do lugar. Diante desse espetáculo celestial, uma tribo de nômades, tocada pela grandiosidade do evento, encontrou os destroços reluzentes e decidiu erguer ali sua morada, ligando em sete pontas uma muralha, abençoando a região com o nome Vallen. Os fragmentos celestiais transformaram a floresta em um lugar mágico, gerando frutos e transformando-a em um paraíso na terra. Ao longo de séculos os nômades formaram um vilarejo com a missão de proteger os cristais, garantindo a harmonia e a prosperidade da Floresta abençoada.

Otavius, um habitante do vilarejo, era um cientista fascinado pelos cristais, firmemente convicto de seu potencial extraordinário. Sua mente inquieta e ambiciosa o levou a substituir um dos cristais por uma pedra semelhante para seu uso pessoal, sem que a aldeia percebesse. Ao aplicá-la em um colar, ele descobriu que conseguia aumentar sua força e vitalidade.

Você, um jovem morador da aldeia, está caminhando e observa algo estranho acontecendo, decide

- [1] Investigar para ver se tem algo rolando realmente
- [2] Deixar para lá e continuar a caminhar

Seu coração acelera e a tensão se instala, pois uma decisão crucial deverá ser tomada:

 Jogador
 Vida: 22
 Força: 12
 Agilidade: 7

Sarah
 Vida: 22
 Força: 12
 Agilidade: 7

Otavius
 Vida: 46
 Força: 27
 Agilidade: 4

- [1] Confrontar Otavius diretamente:
 Arrisque uma tentativa de combate e tente vencer Otavius.
- [2] Tentar negociar:
 Arrisque uma tentativa de negociar com Otavius, buscando convencê-lo a desistir de seus planos malignos.

Vocês olham ao redor e veem alguns objetos que podem ser úteis para enfrentar Otavius, e você decide usar:

- [1] Atacar com pedaço de madeira afiado.
- [2] Atacar com uma pedra pesada.

BIBLIOGRAFIA:

Na elaboração deste trabalho, foram consultados os seguintes recursos:

BaseExemploJOGO.zip: Este recurso consiste em um arquivo fornecido pelo professor durante o curso como exemplo prático para orientar o desenvolvimento do projeto. Ele serviu como referência para entender a estruturação e implementação de um projeto similar. (Arquivo adquirido pelo Teams)

Ragnarok: Informações foram obtidas do site oficial do jogo Ragnarok Online, especificamente da seção que detalha as classes de personagens disponíveis no jogo. (<https://playragnarokonlinebr.com/classes>)

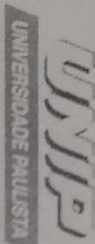
The Witcher: Referências foram extraídas do universo de The Witcher, uma franquia de jogos de RPG bem conhecida. Essas referências foram utilizadas para inspirar elementos narrativos. (<https://www.thewitcher.com/us/pt-br/>)

UNIVERSIDADE PAULISTA

CÓDIGO DA ATIVIDADE: 1659

(1) Horas atribuídas de acordo com o regulamento das Atividades Supervisionadas do curso

CARIMBO E ASSINATURA DO COORDENADOR DO CURSO



FICHA DAS ATIVIDADES PRÁTICAS SUPERVISIONADAS - APS

NAME: Theresa Leana Emerald TURMA: CC2 P04 RA: G85CEA5

CURSO: Ciência da Computação CAMPUS: Paulista
SEMESTRE: 2º TURMA: Nobrega

CÓDIGO DA ATIVIDADE: 7639 SEMESTRE: 2º ANO GRADE: 2024

DATA DA ATIVIDADE	DESCRIÇÃO DA ATIVIDADE	TOTAL DE HORAS	ASSINATURA DO ALUNO	HORAS ATRIBUÍDAS (1)	ASSINATURA DO PROFESSOR
22.05.22	Discussão de textos	15h	Luís		
27.05.22	Pesquisa e codificação	10h			
27.05.22	Codificação	20h			
22.05.22	Documentação	30h			

(1) Horas atribuídas de acordo com o regulamento das Atividades Supervisionadas do curso.

TOTAL DE HORAS ATRIBUIDAS: _____

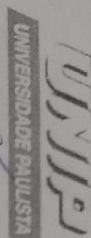
AValiação: _____

Aprovado ou Reprovado

NOTA: _____

DATA: / /

CARIMBO E ASSINATURA DO COORDENADOR DO CURSO



FICHA DAS ATIVIDADES PRÁTICAS SUPERVISIONADAS - APS

NOME: Alison de Carvalho Silva TURMA: CC2104 RA: 100493-2
 CURSO: Licenciatura em Letras CAMPUS: Paulista SEMESTRE: 2 TURNO: Noite
 CÓDIGO DA ATIVIDADE: 7GB9 SEMESTRE: 2 ANO GRADE: 2024

DATA DA ATIVIDADE	DESCRIÇÃO DA ATIVIDADE	TOTAL DE HORAS	ASSINATURA DO ALUNO	HORAS ATRIBUÍDAS (1)	ASSINATURA DO PROFESSOR
27/05	Discussão de Ideias	15h	<i>Alison</i>		
27/05	Resumo Codificação	10h			
27/05	Codificação	20h			
22/05	Documentação	30h			

(1) Horas atribuídas de acordo com o regulamento das Atividades Práticas Supervisionadas do curso.

TOTAL DE HORAS ATRIBUÍDAS: _____

AValiação: _____

Aprovado ou Reprovado

NOTA: _____

DATA: ____/____/____

CARIMBO E ASSINATURA DO COORDENADOR DO CURSO

TURNO: CC2 PD4

SEMESTRE: 2

SEMESTRE: 2^o

AND GRADE: 2024

(1) *Horas atribuídas de acordo com o regulamentamento das Atividades Práticas Supervisionadas do curso.*

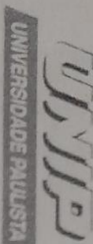
AVALIAÇÃO:

Aprovado ou Reprovado

MOTTA-

DATE: / /

CARIMBO E ASSINATURA DO COORDENADOR DO CURSO



FICHA DAS ATIVIDADES PRÁTICAS SUPERVISIONADAS - APS

NOME: Maria Eduarda R. B. B. dos Santos TURMA: CC 2004 RA: F3561C6
CURSO: Ciência da Computação PAULISTA SEMESTRE: 2º TURNO: Nocturno
CÓDIGO DA ATIVIDADE: 7639 SEMESTRE: 2º ANO GRADE: 2024

DATA DA ATIVIDADE	DESCRIÇÃO DA ATIVIDADE	TOTAL DE HORAS	ASSINATURA DO ALUNO	HORAS ATRIBUÍDAS (1)	ASSINATURA DO PROFESSOR
22.05 a 27.05	Discussão de Advers	15h	<i>[Assinatura]</i>		
	Desenvolvimento Programa	10h			
	Codificação	30h			
	Documentação X	20h			

(11) Horas atribuídas de acordo com o Regulamento das Atividades Práticas Supervisionadas do curso.

TOTAL DE HORAS ATRIBUÍDAS: _____

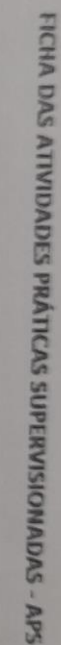
AValiação: _____

Aprovado ou Reprovado

NOTA: _____

DATE: / /

CAMBIO E ASSINATURA DO COORDENADOR DO CURSO



TURNO: CC2804 PA: F35302-8

SEMESTRE: 2º TURNO: Nocturno

ANNO GRADE: 2024

[illegible]

(11) Horas atribuídas de acordo com o regulamento das Atividades Práticas Supervisionadas do curso.

TOTAL DE HORAS ATRIBUÍDAS:

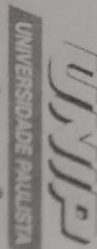
AVALIAÇÃO:

Aprovado ou Reprovado

NOTA:

DATA: ____/____

CARIMBO E ASSINATURA DO COORDENADOR DO CURSO



FICHA DAS ATIVIDADES PRÁTICAS SUPERVISIONADAS - APS

NOME: duin guido do N. Oliveira

TURKISH: CC2004 RA: P013421

curso: ciência da computação campus: Paulista

SEMESTRE: 2º TURNO: Noturno

CÓDIGO DA ATIVIDADE: 76B9

SEMESTRE: 2º

AND GRADE: 2024

DATA DA ATIVIDADE	DESCRIÇÃO DA ATIVIDADE	TOTAL DE HORAS	ASSINATURA DO ALUNO	HORAS ATRIBUÍDAS (1)	ASSINATURA DO PROFESSOR
20.05.2022	Discussão de Ideias	15h	Guilherme		
27.05	Pesquisa Codificação	20h			
22.05.2022	Codificação	40h			

(1) Horas atribuídas de acordo com o regulamentamento das Atividades Supervisionadas do curso.

TOTAL DE HORAS ATRIBUÍDAS:

AVALIAÇÃO:

Aprovado ou Reprovado

MOTTA:

DATA: ____/____/____

CAPIRIBO E ASSINATURA DO COORDENADOR DO CURSO

FICHA DAS ATIVIDADES PRÁTICAS SUPERVISIONADAS - APS

UNIVERSIDADE PAULISTA

NAME: Wang, Chen

TURMA: CC2P09

PA: R022FF3

CURSO: Ciência da Computação Poulista

SEMESTRE: 2^o

TURNO: Noite

CÓDIGO DA ATIVIDADE: 76B9

SEMESTRE: 20

AND GRADE: 2024

DATA DA ATIVIDADE	DESCRIÇÃO DA ATIVIDADE	TOTAL DE HORAS	ASSINATURA DO ALUNO	HORAS ATRIBUÍDAS (1)	ASSINATURA DO PROFESSOR
22.05 a 27.05	Discussão de Advers	15h	Willington		
	Pesquisa Codificação	20h			
	Codificação	40h			

(1) Horas atribuídas de acordo com o regulamento das Atividades Práticas Supervisionadas do curso.

TOTAL DE HORAS ATRIBUÍDAS:

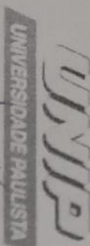
AVALLAÇAO:

Aprovado ou Reprovado

PHOTA:

DATE: / /

CARIMBO E ASSINATURA DO COORDENADOR DO CURSO



FICHA DAS ATIVIDADES PRÁTICAS SUPERVISIONADAS - APS

NAME: Northern Davis Concordia

TURPANA: CC2PO4 RA: R022FD7

CURSO: Ciência da computação CAMPUS Pouloista

SEMESTRE: 2º TURNO: Noturno

CÓDIGO DA ATIVIDADE: 76B9

SEMESTRE: 2º

AND GRADE: 2024

DATA DA ATIVIDADE	DESCRIÇÃO DA ATIVIDADE	TOTAL DE HORAS	ASSINATURA DO ALUNO	HORAS ATRIBUÍDAS (1)	ASSINATURA DO PROFESSOR
22.05 a 27.05	Discussão de Ideias	15h	At		
	Projeto Codificação	20h			
	Codificação	40h			

(1) Horas atribuídas de acordo com o regulamentamento das Atividades Práticas Supervisionadas do curso.

TOTAL DE HORAS ATRIBUÍDAS:

AVALIAÇÃO:

Aprovado ou Reprovado

NOTA: _____

DATE: / /

CARIMBO E ASSINATURA DO COORDENADOR DO CURSO