

Trabalho Prático 01 – AEDS 1 – Trios

Professora: Thais R. M. Braga Silva

Valor: 15 pontos

Data de Entrega (código + documentação): 26/09/2023

Data das Entrevistas: 03 e 05/10/23 (nas aulas de laboratório)

Forma de Entrega: PVANet (formato .zip ou .tar.gz)

Neste trabalho, vamos implementar uma versão de terminal do jogo Paciência, utilizando listas lineares e TADs. Paciência é um jogo de cartas de apenas um jogador, que se tornou extremamente popular por já vir instalado com vários sistemas operacionais. O jogo possui algumas variações, e a que deve ser implementada é a chamada [Klondike](#), presente em muitas versões do Microsoft Windows, e portanto bem conhecida por muitas pessoas. A jogabilidade será feita apenas através do terminal, portanto não é necessário criar interface gráfica.

O jogo usa um baralho convencional de 52 cartas, contendo 4 naipes – espadas, ouros, paus e copas –, e 13 valores distintos – Ás (A), Valete (J), Rainha (Q), Rei (K), e os números de 2 a 10. A ordem desses valores é, do menor ao maior: A, 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K. As cartas são dispostas seguindo um procedimento específico para dar início a uma partida (Figura 1).

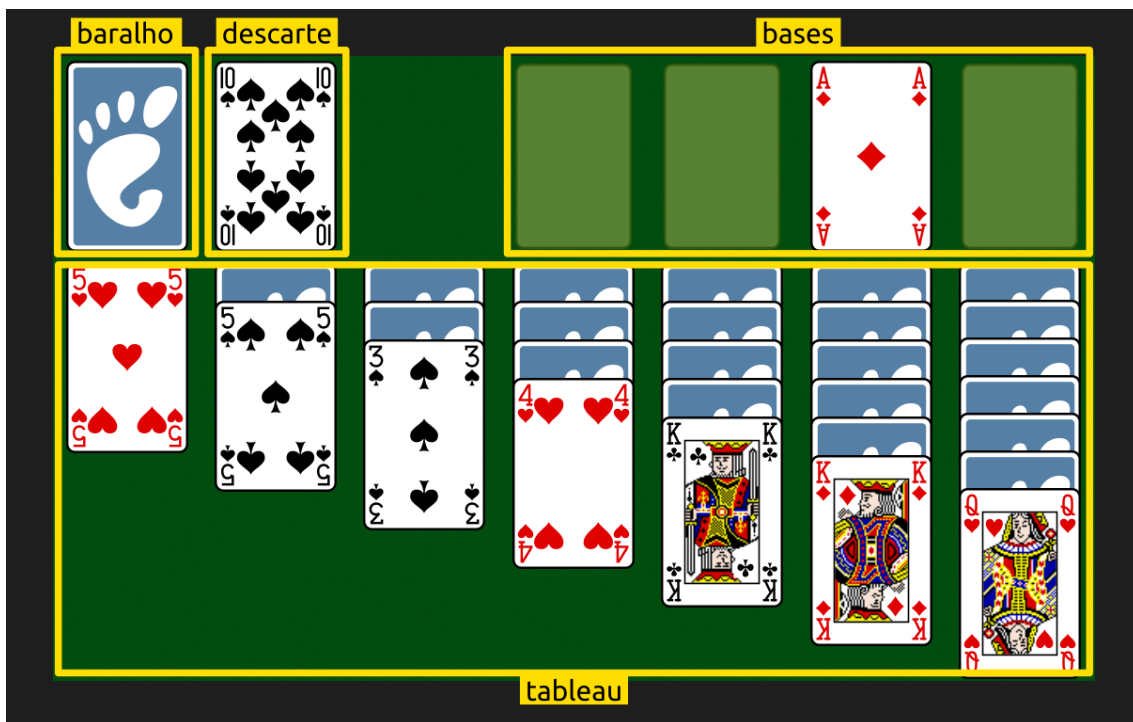


Figura 1. Disposição das cartas num jogo de Paciência Klondike e o nome de cada espaço.

Primeiramente embaralhamos as cartas e as colocamos viradas para baixo na posição indicada para o **baralho**. Ao lado do baralho, teremos o **descarte**, que começa vazio. Abaixo do baralho, temos 7 colunas de cartas no espaço chamado **tableau**. Na 1ª coluna é colocada 1 carta. Na 2ª

coluna são colocadas 2 cartas. Na 3ª coluna, 3 cartas. E assim sucessivamente até a 7ª coluna. Todas as cartas são colocadas viradas para baixo, com exceção das últimas cartas em cada coluna, que são reveladas, viradas para cima. Dizemos que essas cartas estão no topo da coluna. Acima do tableau, temos as quatro **bases**, onde devemos juntar as cartas para ganhar o jogo. As bases se iniciam vazias, e há uma base reservada para cada naipe.

No **tableau**, as cartas viradas para cima devem sempre respeitar a ordem das cartas, alternando suas cores, sendo as cartas de menor valor colocadas sobre as cartas de maior valor. Os Reis, que são as cartas de maior valor, só podem ser colocados em cima de espaços vazios no tableau. Toda vez que uma carta virada para baixo no tableau se torna o novo topo de sua coluna, essa carta deve ser virada para cima, e dizemos que a carta foi revelada.

Exemplos: na Figura 1, é possível colocar o **4 de Copas** sobre o **5 de Espadas**, e a carta abaixo do **4 de Copas** seria revelada. Porém não podemos fazer isso para o **5 de Copas**, pois ambas são cartas de naipe vermelho. Já os Reis não poderiam ser colocados sobre nenhuma outra carta até que se abra um espaço vago.

Nas **bases**, as cartas devem ser colocadas também em sequência, porém na ordem inversa, com as cartas de maior valor por cima das cartas de menor valor. Portanto, os Áses devem iniciar a sequência. Além disso, todas as cartas em uma base devem ser do mesmo naipe.

Exemplo: na Figura 1, o **Ás de Ouros** já iniciou uma sequência, enquanto as demais bases estão vazias ainda, esperando seus respectivos Áses. Quando o **2 de Ouros** aparecer, ele poderá ser colocado por cima do **Ás de Ouros**.

No jogo há seis movimentos válidos, com suas respectivas abreviações:

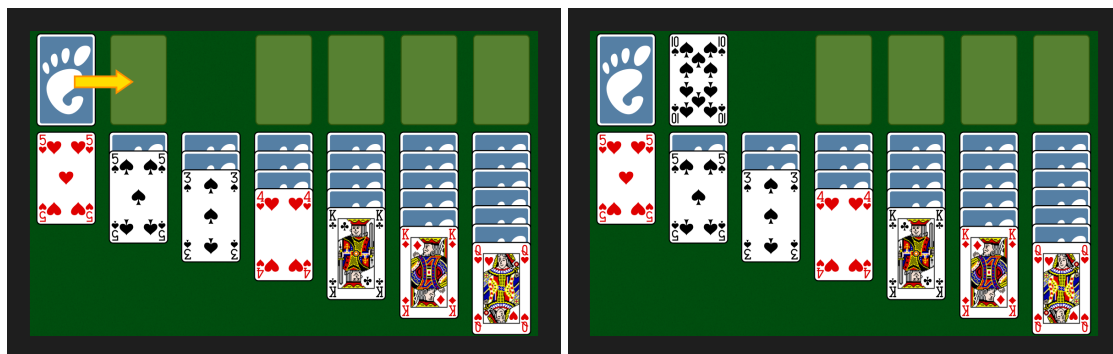


Figura 2. Exemplo da ação de comprar uma carta.

- **Comprar uma carta (CC) (Figura 2):** consiste em virar uma carta do **baralho** para o **descarte**, com a face para cima. Se não houver cartas no baralho, ao invés disso reciclamos o baralho, ou seja, todas as cartas do descarte são colocadas uma a uma de volta para o baralho, viradas para baixo.

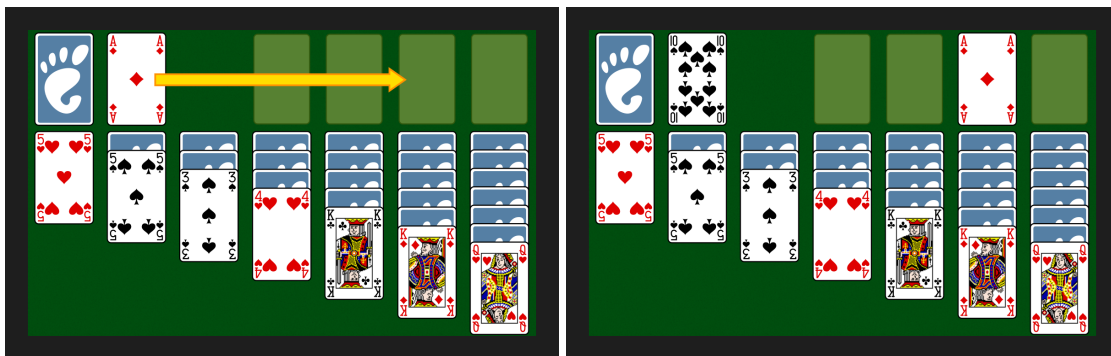


Figura 3. Exemplo da ação de mover do descarte para as bases.

- **Mover do descarte para as bases (DB) (Figura 3):** consiste em mover a carta que estiver no topo do **descarte** para a sua respectiva **base**, se possível.

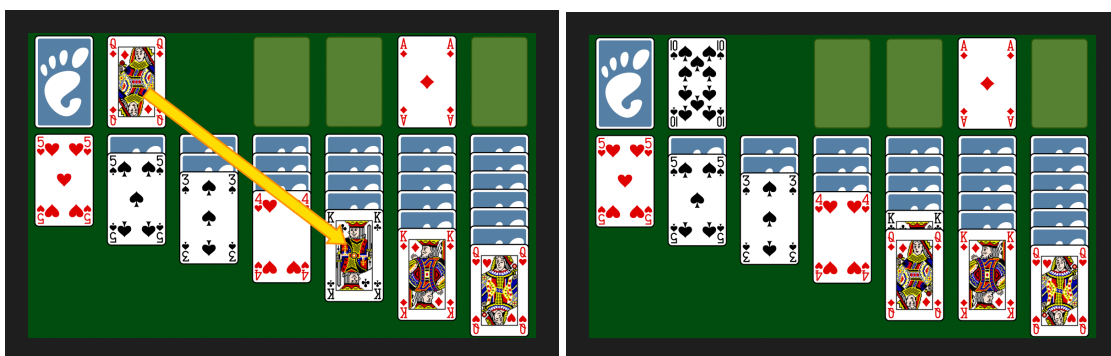


Figura 4. Exemplo da ação de mover do descarte para o tableau.

- **Mover do descarte para o tableau (DT) (Figura 4):** consiste em mover a carta que estiver no topo do do **descarte** para o topo de alguma coluna do **tableau**, se possível.

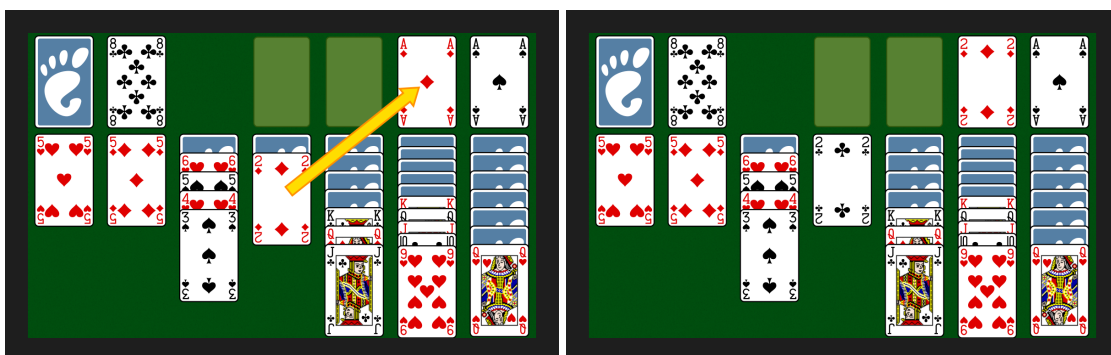


Figura 5. Exemplo da ação de mover do tableau para as bases.

- **Mover do tableau para as bases (TB) (Figura 5):** consiste em mover uma carta do topo de alguma coluna do **tableau** para sua respectiva **base**, se possível.

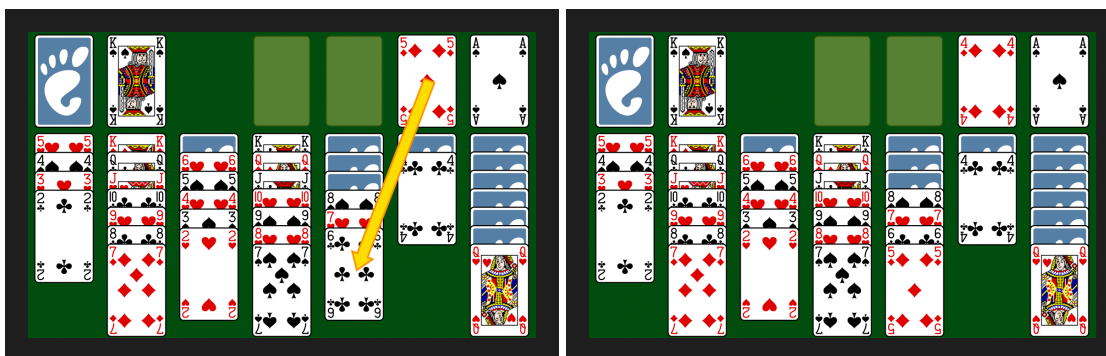


Figura 6. Exemplo da ação de mover das bases para o tableau.

- **Mover das bases para o tableau (BT) (Figura 6):** consiste em mover uma carta do topo de alguma **base** para o topo de alguma coluna do **tableau**, se possível.

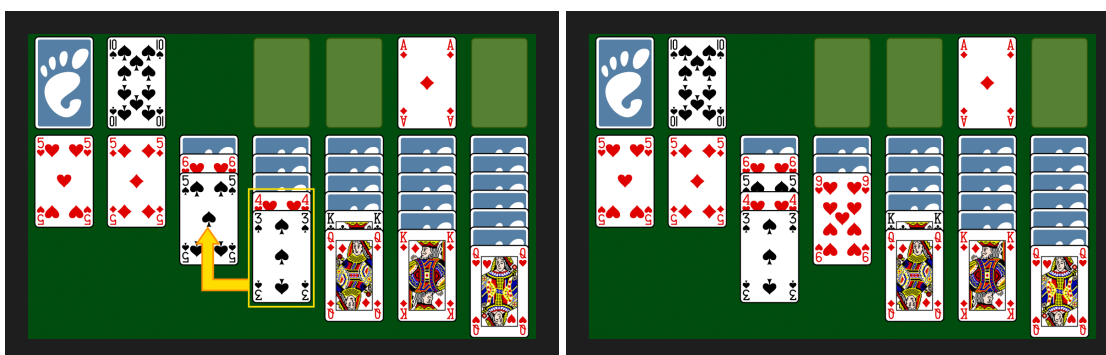


Figura 7. Exemplo da ação de mover entre colunas do tableau.

- **Mover entre colunas do tableau (TT) (Figura 7):** consiste em mover uma quantidade arbitrária de cartas a partir do topo de uma coluna do **tableau** para outra coluna, se possível. A ordem das cartas é preservada, e são movidas todas juntas.

O jogo encerra quando todas as cartas estiverem colocadas em suas devidas bases, mas nem sempre é possível ganhar. Além disso, é mantida uma pontuação para avaliar o desempenho do jogador, como mostra a Tabela 1. A pontuação é sempre no mínimo zero.

Movimento	Pontuação
Reciclar o baralho	-50
Descarte para base	+10
Descarte para tableau	+5
Tableau para base	+10
Base para tableau	-15
Revelar carta do tableau	+5

Tabela 1. Pontuação do jogo.

A estrutura de dados a ser utilizada para a implementação das listas lineares será OBRIGATORIAMENTE LISTA ENCADEADA (em todos os lugares em que uma lista for implementada).

Abaixo estão descritos, em detalhes, quais TADs deverão ser implementados neste trabalho, com os dados que devem estruturar e suas operações. É esperado que os alunos façam implementações que sigam completa e corretamente estas especificações.

TIPO ABSTRATO DE DADOS CARTA

O primeiro TAD a ser implementado deve ser o TAD Carta. Este TAD representa uma carta dentro do jogo, e portanto sua estrutura deve conter os campos:

- naipe (copas, espadas, ouros, paus);
- valor (A, 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K);
- posição (para cima ou para baixo).

Uma carta tem seu naipe e valor atribuídos uma única vez, e depois esse valor não muda. Já sua posição pode ser mudada ao longo do jogo. Além disso, esse TAD será responsável por verificar a relação entre as cartas, ou seja, se duas cartas consecutivas formam uma sequência de mesmo naipe, ou uma sequência de cor alternada. Assim, as operações do TAD são:

- Criar: inicializa a carta com seu naipe, valor e posição inicial.
- Retornar naipe;
- Retornar valor;
- Retornar posição;
- Alterar posição;
- Verificar sequência de mesmo naipe: recebe duas cartas como parâmetro, e retorna verdadeiro caso a segunda carta seja adequada como próxima carta numa sequência de mesmo naipe para a primeira carta. Obs.: se a primeira carta é nula, então o valor válido para a segunda carta é um Ás;
- Verificar sequência alternada: recebe duas cartas como parâmetro, e retorna verdadeiro caso a segunda carta seja adequada como próxima carta numa sequência de cores alternadas para a primeira carta. Obs.: se a primeira carta é nula, então o valor válido para a segunda carta é um Rei;
- Exibir: imprime uma representação da carta, levando em conta que se a carta estiver virada para baixo, seu valor e naipe não devem aparecer. A forma como a carta é representada no terminal é definida pelo grupo, sendo possível utilizar caracteres simples, caracteres especiais, cores, etc.

TIPO ABSTRATO DE DADOS LISTA DE CARTAS

A finalidade deste TAD é gerenciar uma lista de cartas, para representar cada um dos espaços onde podemos colocar cartas na mesa do jogo. Em todos esses espaços, é importante saber qual carta está no topo (puxamos cartas do topo do baralho, inserimos cartas no topo das colunas do tableau, etc.).

Assim, na implementação da lista encadeada, o grupo deve definir qual das pontas da lista se refere ao topo da lista de cartas. Por exemplo, uma opção é definir o início da lista como sendo o topo daquele conjunto de cartas. Além da lista encadeada de cartas, deve ser colocado no TAD um campo que registra o número de cartas na lista. As operações a serem disponibilizadas devem ser:

- Criar: inicializa a lista vazia;
- Retornar tamanho;
- Verificar se está vazia;
- Retornar a carta que está no topo;
- Retornar a carta que está numa dada posição: a posição **0** se refere à carta que está no topo, a posição **n-1** se refere à carta que está no fundo, sendo **n** o tamanho da lista;
- Adicionar carta ao topo;
- Retirar carta do topo.
- Transferir cartas: transfere uma quantidade arbitrária de cartas a partir do topo de uma lista, para o topo de outra, mantendo a ordem original.
- Embaralhar: faz uma permutação aleatória da lista, alterando a mesma.
- Exibir: imprime o conteúdo da lista de cartas, sendo necessário indicar quando a lista está vazia. Além disso, deve haver um parâmetro para indicar se todas as cartas devem ser exibidas, ou apenas a carta do topo.

TIPO ABSTRATO DE DADOS MESA

Este TAD deve representar uma mesa do jogo Paciência, contendo as listas de cartas para cada espaço, e a contagem dos pontos. Assim, a estrutura deste TAD deve conter os campos descritos abaixo. Obs.: para as bases e para o tableau, é recomendado que se use vetores.

- baralho: uma lista de cartas;
- descarte: uma lista de cartas;
- bases: quatro listas de cartas, uma para cada naipe;
- tableau: sete listas de cartas;
- pontuação.

As operações do TAD englobam todas as ações necessárias para fazer o jogo funcionar. Este TAD é que chama as operações do TAD Carta para conferir se as jogadas são válidas.

- Inicializar: inicializa cada uma das listas de cartas como vazias e zera a pontuação.
- Carregar baralho aleatório: insere todas as 52 cartas no baralho, e o embaralha.
- Carregar baralho: insere cartas no baralho de acordo com as cartas passadas num parâmetro de entrada, preservando a ordem passada.
- Preparar: distribui as cartas do baralho para o tableau, como descrito na introdução.
- Verificar vitória: confere se a condição de vitória do jogo foi atingida;
- Exibir: imprime a disposição das cartas na mesa, chamando a função de exibir lista de cartas adequadamente. A forma exata de como exibir a mesa é de escolha livre do grupo. Não há problema em imprimir as colunas do tableau como linhas, na horizontal, visto que seria muito trabalhoso imprimir as listas em colunas lado a lado como mostram as Figuras 1 a 7. A pontuação também deve ser exibida.

- Comprar carta: executa a ação de comprar uma carta.
- Mover do descarte para as bases;
- Mover do descarte para o tableau: tem como parâmetro o índice de qual coluna do tableau vai receber a carta;
- Mover do tableau para as bases: tem como parâmetro o índice de qual coluna do tableau vai ser obtida a carta;
- Mover das bases para o tableau: tem como parâmetro o índice de qual base vai ser obtida a carta, e o índice de qual coluna do tableau vai receber essa carta;
- Mover entre colunas do tableau: tem como parâmetro a quantidade de cartas a ser movidas, o índice de qual coluna do tableau vão ser obtidas as cartas, e o índice de qual coluna do tableau vai receber as cartas.

SISTEMA DE TESTE

Para testar a implementação dos seus TADs, o grupo deverá implementar um programa principal que representa o sistema do jogo Paciência. Para tanto, este programa deverá possuir as seguintes funcionalidades:

- Inicialização: inicializa uma variável do TAD Mesa.
- Seleção de dois modos de utilização do sistema: interativo e por arquivo. No modo interativo, o baralho deve ser aleatório e o usuário vai escolher manualmente cada ação que deseja realizar no jogo. No modo arquivo, o usuário informa o caminho do arquivo a ser carregado, o qual estará formatado linha a linha com o baralho inicial que deve ser carregado e parâmetros a serem executados.
- Para o modo interativo, o programa entra em um loop, sempre exibindo o estado atual da mesa do jogo, e um menu de opções resumindo as seis jogadas possíveis, além de uma opção para encerrar o programa. Se a condição de vitória for atingida, essa informação deve ser exibida e o programa encerra imediatamente.
- Para o modo arquivo, deve ser exibido o estado inicial do jogo após as cartas serem distribuídas no tableau, e o estado final do jogo, após todas as operações. Deve constar também se foi atingida a condição de vitória no estado final.

Todos os resultados deverão ser exibidos na saída padrão.

ARQUIVO DE ENTRADA

Para o modo de utilização por arquivo, o arquivo de entrada deverá possuir o seguinte formato. Na primeira linha, há um número inteiro entre 28 e 52, indicando quantas cartas serão inseridas no baralho. Em seguida, cada carta é listada, separada por caracteres em branco, de forma que a última carta listada será a carta do topo. A entrada de cada carta segue o formato **(N V)**, no qual N é o naipe, sendo um caractere C, P, O, ou E, indicando Copas, Paus, Ouros ou Espadas, e V é um número inteiro entre 1 e 13.

Em seguida, são listadas as operações, uma em cada linha, começando com a abreviação de uma possível ação (CC, DB, DT, TB, BT, ou TT), seguida opcionalmente por parâmetros separados por espaços. No caso das ações DT e TB, é passado o índice da coluna do tableau a ser manipulada. No caso da ação BT, é passado um caractere indicando o naipe e um inteiro

indicando o índice do tableau. Para a ação TT, são três números inteiros, sendo o primeiro a quantidade de cartas a ser movida, o segundo é o índice da coluna de origem, e o terceiro é o índice da coluna de destino. Todos os índices são números entre 1 e 7.

TESTES

Será disponibilizado junto com a especificação deste trabalho um pequeno arquivo de teste e as saídas esperadas para o mesmo. O grupo deve, no entanto, criar seus próprios casos de teste, tanto interativos como por arquivo. Novos testes por arquivo serão utilizados no dia da entrevista, a ser realizada como parte da avaliação deste trabalho.

Em particular, atente para:

- O programa deve ser organizado em módulos, conforme estudado em sala de aula. O módulo do programa principal deve estar separado dos módulos que compõem os TADs.
- O programa deve estar bem indentado e comentado.
- Caso apareçam números fixos no código, estes devem ser definidos como constantes.
- **Trabalhos copiados serão penalizados.**

ATENÇÃO: Soluções que não correspondam à implementação de Tipos Abstratos de Dados serão duramente penalizadas por não atenderem à especificação. A especificação pode conter erros ou problemas não intencionais que podem e devem ser reportados, caso encontrados. Além disso, diversas decisões de projeto e desenvolvimento precisam e devem ser feitas por cada trio. Implementações extras são possíveis e serão bem vindas, desde que tudo o que foi explicitamente pedido nesta especificação tenha sido contemplado, e que o objetivo central do tema do trabalho não seja comprometido.

O que deve ser entregue:

- Todo código fonte produzido, incluindo os arquivos de cabeçalho
- Uma pequena porém completa documentação, utilizando o formato disponibilizado pela professora. O formato para a entrega da documentação é pdf
- Fazer um zip ou tar.gz de todos os arquivos, nomeá-lo com o nome e número de matrícula do grupo, bem como número do TP, e submetê-lo apenas uma vez pelo PVANet Moodle (NÃO UTILIZAR COMPACTAÇÃO RAR!).

Como será a avaliação:

- Entrevista com os monitores
- Avaliação do código + documentação (monitores e professora)