

Relatório Trabalho 2

Fodinha

Redes de Computadores 1
Universidade Federal do Paraná – UFPR
André Gauer Thomal
Rafael Gonçalves dos Santos
Curitiba – PR

I. Introdução

Este presente relatório tem como objetivo discorrer sobre as principais decisões tomadas para a realização do Trabalho 2.

II. Inicialização e Regras

1. Inicialização

Para garantir um correto início, cada pessoa deve se identificar usando um número de 0 a 3 durante a inicialização do programa, como por exemplo:
\$./fodinha 0

2. Regas

Cada jogador começa com 5 cartas e, a cada rodada, o número de cartas em mãos diminui conforme são jogadas. O jogo continua até que um dos participantes perca todas as suas 12 vidas.

III. Sobre o Jogo

1. Execução

O sistema de comunicação do jogo inicia com um jogador que assume o papel de carteador, responsável por comandar as ações daquela rodada. Enquanto isso, os demais jogadores apenas recebem as mensagens, processam-nas e as transmitem adiante. Essa relação se mantém até o final da partida, que ocorre quando um jogador perde todas as suas vidas.

2. Token

4 by	4 by	4 by	16 by	16 by	4 by	1024 by
status	src	addr	dest	recv	type	data

Formato do token

status: integridade do envio

src: remetente do envio

addr: endereço atual do token

dest: destinos da mensagem

recv: verificações de recebimento

type: assunto da mensagem

data: conteúdo da mensagem

3. Envio

Cada jogador tem sua própria porta de comunicação. Sempre que um jogador recebe uma mensagem, ele a envia para a mesma porta, que corresponde ao próximo jogador na sequência.

4. Recebimento

Quando um jogador recebe uma mensagem, se ele não conseguir ler a mensagem, ele define esse status no campo *recv* do token e a envia para o próximo jogador. Se ele conseguir ler a mensagem, ele define esse status no campo *recv* do token, trata a mensagem de acordo com o valor do campo *type* e então envia para o próximo jogador.

5. Ciclo de envios e recebimentos

Jogadores comuns apenas recebem e enviam o token para o próximo jogador. O carteador, por outro lado, envia o token e aguarda seu recebimento (após o token dar a volta completa no anel) para verificar o status da mensagem. Se o status da mensagem for negativo, ele reenvia a mensagem; se for positivo, ele apenas continua o jogo.