

Sistemas Distribuídos

SPHINX, O JOGO

Aluno:

Eduardo Stuart Napier Moura

Matrícula: 20161101936

eduardostuart.ti@gmail.com

Professor:

Fábio Contarini Carneiro

Universidade Veiga de Almeida
2018.2



Sphinx

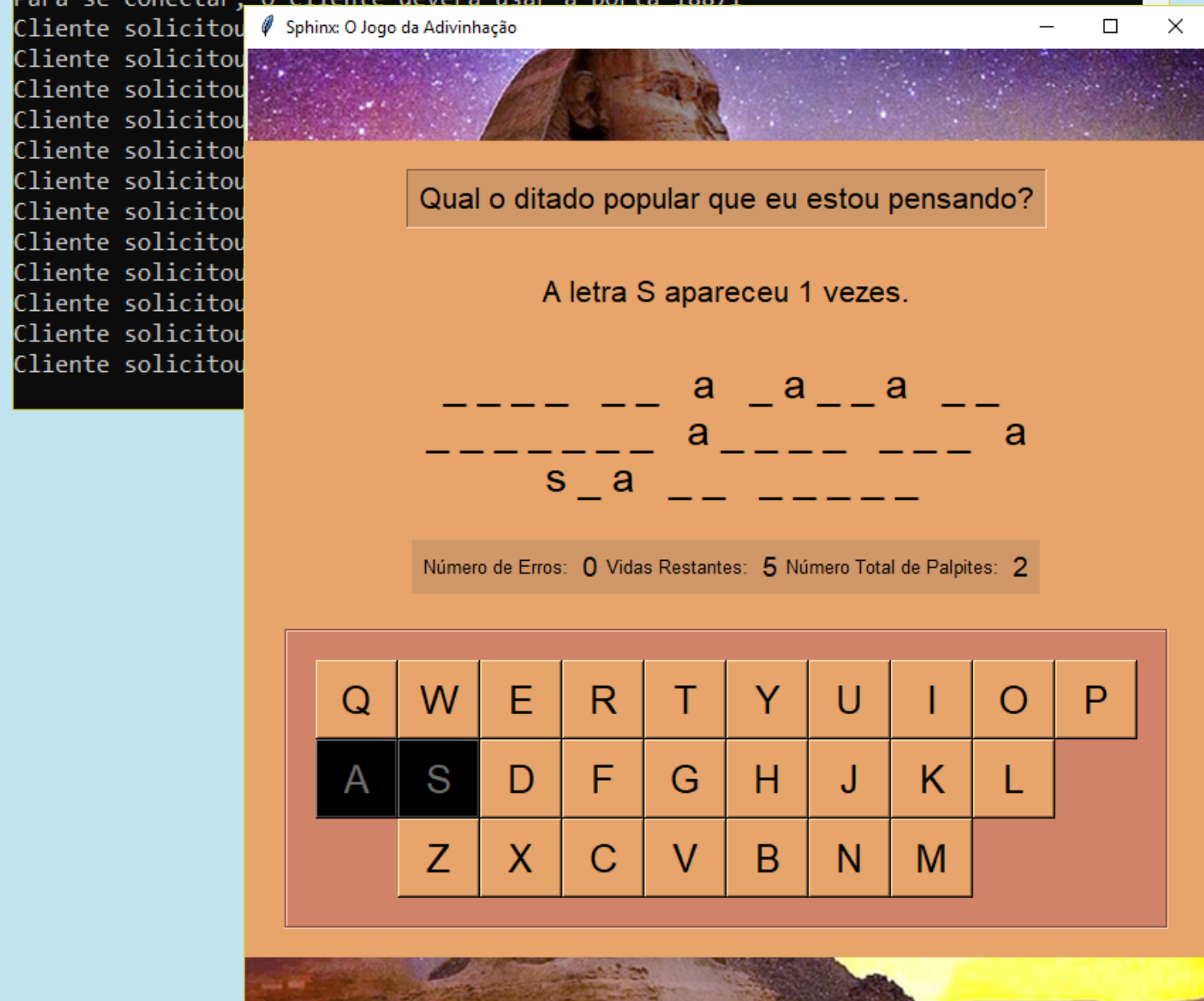
PROPOSTA

- Implementar um jogo que esteja rodando em um sistema distribuído
- Empregar uma linguagem que apresenta alta portabilidade e que permita a alteração do código-fonte sem a necessidade de recompilar o projeto
- Usar o paradigma Cliente-Servidor, em que o servidor se encarrega de controlar o estado do jogo, enquanto o programa cliente promove a interface com o jogador

```
Prompt de Comando - python SphinxServer.py
C:\bt\SistemasDistribuidos_Trabalho_Sphinx>python SphinxServer.py

#####
##### Sphinx #####
#####
## Trabalho para a disciplina Sistemas Distribuídos ##
## Professor: Fábio Contarini Carneiro ##
## Aluno: Eduardo Stuart Napier Moura (eduardostuart.ti@gmail.com) ##
## Matrícula: 20161101936 ##
## Universidade Veiga de Almeida - Período: 2018.2 ##
#####
#####

Cliente solicitou: um novo jogo
Seja bem-vindo ao jogo Sphinx
O servidor está em operação
Para se conectar, o cliente deverá usar a porta 18871
```

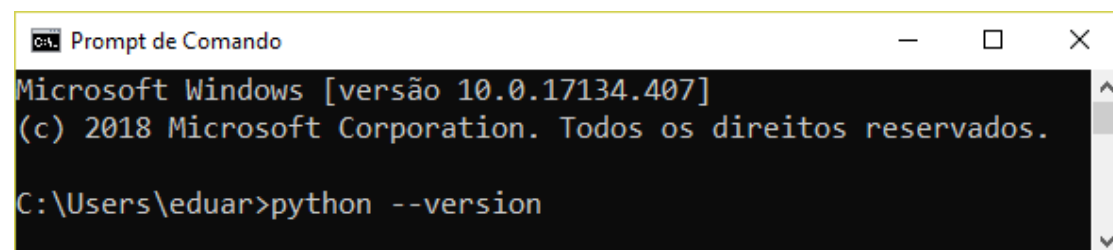


REQUISITOS

Para que o servidor e o cliente possam operar corretamente, será preciso satisfazer dois requisitos: possuir o interpretador **Python** instalado (versão 3.6 ou superior) e instalar, via o comando **pip**, a biblioteca que irá prestar os serviços de **RPC** (*remote procedure call*) para o nosso sistema.

Para testar se o Python está instalado e qual a sua versão, é simples:

- Ative o *prompt* de comando do seu sistema operacional. Nas imagens a seguir, usaremos o Windows como referência, mas o procedimento é similar tanto para o Linux, quanto para o Mac OS.
- Digite o comando **python --version**



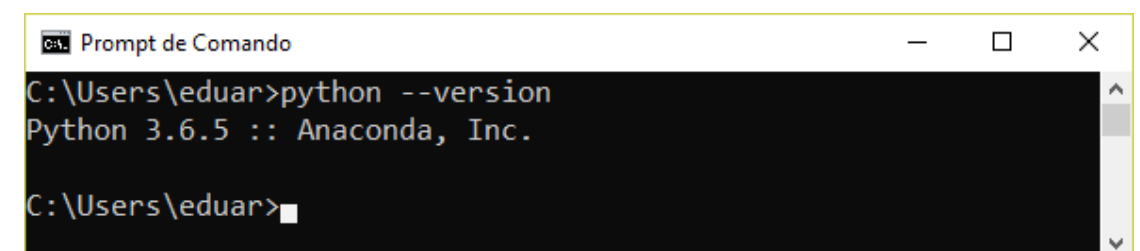
```
C:\Users\eduar>python --version
```

Se for exibido um aviso de erro, indicando que o comando **python** não foi reconhecido, será preciso visitar o site <http://python.org> e, a partir da seção de **Downloads**, seguir os passos para realizar a instalação no seu sistema operacional (observe que, durante o processo inicial de instalação, será perguntando se é desejado incluir o interpretador no **path** do sistema operacional - ative essa opção, isso é importante).

Ao concluir a instalação, não é necessário reiniciar o computador: basta fechar o terminal/prompt de comando e abrí-lo novamente.

Se o comando do passo anterior não apresentar um erro, mas a versão for **inferior a 3.6**, será preciso atualizar o interpretador, visitando o mesmo site citado anteriormente e transferindo a versão mais atual.

Na imagem abaixo, o sistema está com a versão 3.6.5, que é suficiente para a correta execução do jogo.



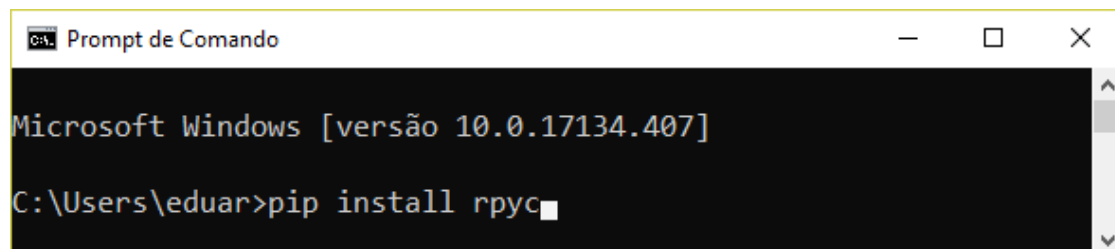
```
C:\Users\eduar>python --version
Python 3.6.5 :: Anaconda, Inc.

C:\Users\eduar>
```


REQUISITOS (CONTINUAÇÃO)

O projeto faz uso da tecnologia RPC para a chamada remota entre processos. Infelizmente, o Python não inclui na instalação padrão o suporte para essa tecnologia, mas é possível incluí-la usando apenas um simples comando.

- Ative novamente a linha de comando/terminal do seu sistema operacional;
- Após testar que o Python está instalado e operante (basta refazer o teste do primeiro passo, se desejar), digite o seguinte comando: **pip install rpyc**



```
Prompt de Comando
Microsoft Windows [versão 10.0.17134.407]
C:\Users\eduar>pip install rpyc
```

Observe que o nome do pacote a ser instalado é **rpyc** (e não somente *rpc* -- o **py** do nome do pacote é uma referência ao Python).

Após a instalação (e assegurando de que não houve nenhum erro), podemos prosseguir para testar o jogo!

SOBRE O SPHINX

Com a proposta de testar os conhecimentos adquiridos sobre **Sistemas Distribuídos e Tolerância de Falhas**, disciplina da **Universidade Veiga de Almeida** e que foi ministrada pelo professor **Fábio Contarini Carneiro**, *Sphinx* foi desenvolvido com os seguintes objetivos:

- Dividir as responsabilidades de um jogo eletrônico ao fazer uso dos sistemas distribuídos, empregando, nesse projeto específico, o paradigma cliente-servidor
- Fazer uso do paradigma **MVC**, sendo que tanto o **modelo**, quanto o **controlador**, estão concentrados no servidor, enquanto o **view** está no cliente.

OBJETIVO

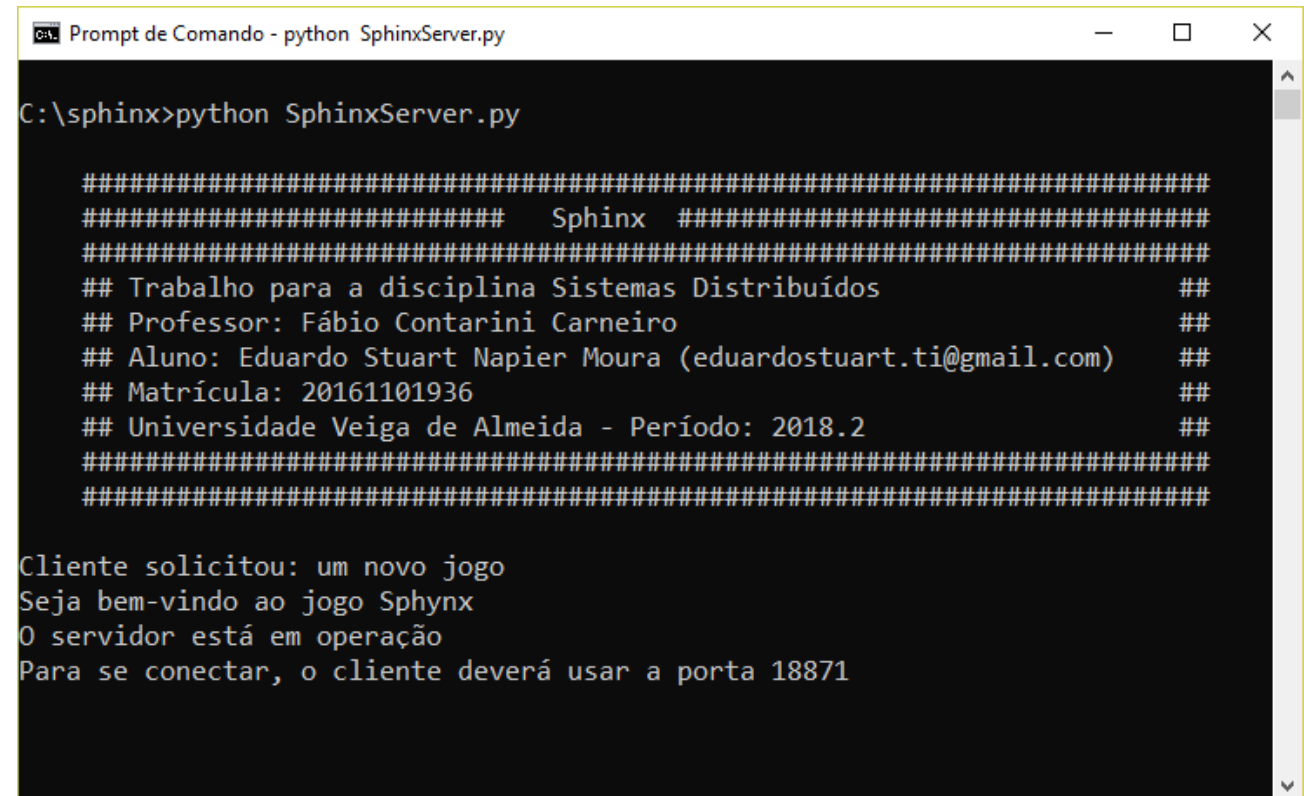
O objetivo do jogo é relativamente simples: desafiar o jogador a acertar um ditado popular, devendo indicar, uma letra por vez, o seu palpite. Porém, o número de erros que podem ser cometidos é limitado, o que faz com que o jogador tenha que usar de alguma astúcia e senso de oportunidade para não ser vencido e ter sua jornada abreviada.

INICIANDO O SERVIDOR

Para que um jogo possa ser iniciado, o servidor precisará estar operando antes do cliente.

Para isso, devemos:

- Descompactar o arquivo fornecido para um mesmo diretório (nesse guia, vamos usar como base o caminho **c:\sphinx**)
- Na linha de comando, devemos mudar para esse diretório (**cd \sphinx**) e executar o seguinte comando: **python SphinxServer.py**



```
Prompt de Comando - python SphinxServer.py

C:\sphinx>python SphinxServer.py

#####
#####   Sphinx   #####
#####
## Trabalho para a disciplina Sistemas Distribuídos      ##
## Professor: Fábio Contarini Carneiro                  ##
## Aluno: Eduardo Stuart Napier Moura (eduardostuart.ti@gmail.com) ##
## Matrícula: 20161101936                                ##
## Universidade Veiga de Almeida - Período: 2018.2      ##
#####
#####

Cliente solicitou: um novo jogo
Seja bem-vindo ao jogo Sphynx
O servidor está em operação
Para se conectar, o cliente deverá usar a porta 18871
```

Ao executar o servidor, uma tela similar a de cima será exibida.

O servidor:

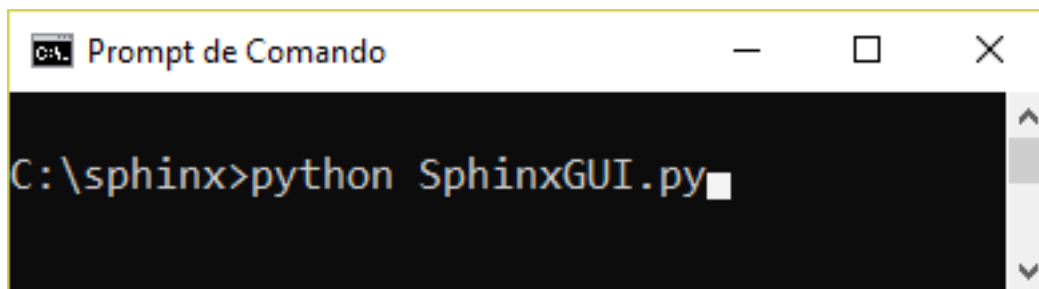
- Está operando na porta **18871** (esse valor pode ser prontamente alterado ao editar o código-fonte e informando uma nova porta)
- Toda e qualquer interação de um cliente irá gerar uma mensagem no console, informando o que está sendo solicitado/fornecido

O CLIENTE

Com o servidor em atividade, podemos iniciar o cliente e iniciar uma partida.

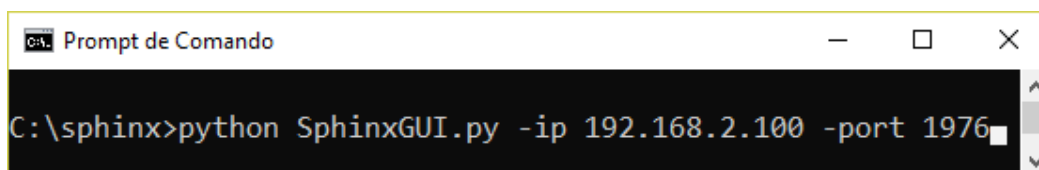
Para ativarmos o cliente, temos duas opções:

- Não fornecermos nenhum parâmetro na linha de comando, que fará com que o cliente assuma que o **IP** destino (ou seja, onde o servidor se encontra) é o *localhost*, e que a porta utilizada é a *default* (**18871**)

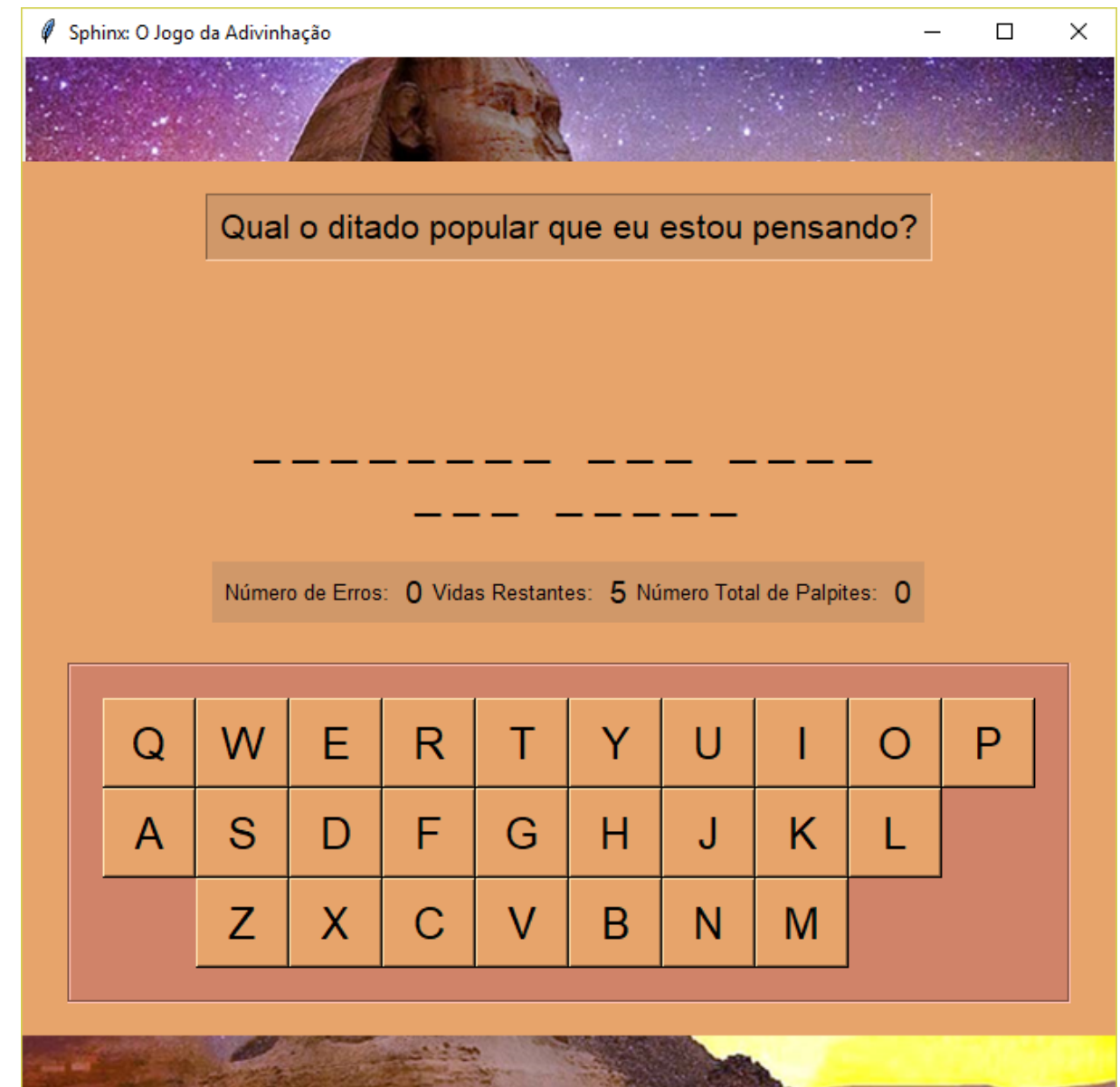


```
C:\sphinx>python SphinxGUI.py
```

- Indicamos o **IP** e/ou a **Porta**, usando, respectivamente, os parâmetros **-ip** e **-port** (no exemplo abaixo, estamos usando o IP **192.168.2.100** e a porta **1976**).



```
C:\sphinx>python SphinxGUI.py -ip 192.168.2.100 -port 1976
```



Ao executar o cliente, será exibida algumas instruções simples sobre a operação do jogo e, assim que o jogador pressionar o botão OK, poderá iniciar uma partida.

Se vencer, ou perder, sempre será dada a oportunidade de tentar desvendar uma nova frase.