Universidade Federal do Ceará
Centro de Ciências
Departamento de Computação

Trabalho Intermediário – Jogo Amedonha
Disciplina Programação (CK0226) – Semestre 2020.2

Prof. Miguel Franklin

***** PARA EQUIPES DE NO MÁXIMO 3 (TRÊS) ALUNOS(AS) *****

Desenvolver um programa em C que implemente o jogo AMEDONHA, baseado no clássico jogo "Adedonha", mas com regras próprias e originais. A implementação deverá ser feita conforme requisitos a seguir:

REQUISITOS FUNCIONAIS

Obs: Como o próprio nome diz, um "Requisito Funcional" é um requisito: é obrigatório. Portanto, a palavra DEVE tem que ser interpretada como em: "Você DEVE comer todo o seu brócolis!", e não como "Deve ter sido o repolho que eu comi ontem."

1. Nesse novo jogo, participam de 2 até 10 jogadores. No início do jogo, o computador DEVE solicitar a indicação da quantidade de jogadores, denominado ***N***. O programa DEVE solicitar também o nome de cada um dos jogadores. Para facilitar a visualização, cada nome poderá ter até 12 caracteres. O programa DEVE validar este limite, solicitando novamente um nome até 12 caracteres caso um maior seja entrado.
2. Em seguida, o computador DEVE sortear uma letra (de A até Z, excluindo K, W e Y).
3. Em cada rodada, o computador DEVE sortear e indicar a categoria, entre "nomes de pessoas", "nomes de cidade", "nomes de animais", "nomes de comida" e "profissões". De uma rodada para outra, as categorias NÃO DEVEM se repetir.
4. Em seguida, o programa DEVE definir, por sorteio, a ordem de resposta dos ***N*** jogadores, que DEVE mudar a cada rodada.
5. O primeiro jogador a responder, a ser chamado pelo nome, terá até (8 + 2 * ***N***) segundos para entrar a sua resposta pelo teclado. O segundo jogador terá 2 segundos a menos. O terceiro, 4 segundos a menos, e assim por diante.
6. Um jogador não é obrigado a utilizar todo o seu tempo para responder. Ao invés disso, pode escolher responder o mais rapidamente possível, para dar menos tempo para pensar ao próximo jogador.
7. Cada resposta deve conter até 30 caracteres. Caso esse limite seja extrapolado, o programa deverá solicitar novamente, e só DEVE terminar a contagem do tempo depois que uma resposta válida for entrada.

8. O programa DEVE verificar se a resposta entrada tem como inicial a letra sorteada. Caso não seja, o programa DEVE solicitar novamente, e só deverá terminar a contagem do tempo depois que uma resposta válida for entrada.
9. O tempo de resposta de cada jogador DEVE começar a ser contado a partir da entrada da resposta do jogador anterior até que ele termine de entrar a sua própria resposta.
10. Caso o jogador não consiga responder dentro do seu tempo, a resposta DEVE ser ignorada.
11. Após a resposta de cada jogador, a tela deve ser limpa para que o programa solicite a jogada do próximo jogador. Um jogador não deve ver a resposta de seus antecessores.
12. As respostas da categoria "nome de pessoa" DEVEM ser de uma só palavra. Isto é, respostas como "José Maria" ou "Antônio Carlos" NÃO DEVEM ser aceitas. Caso o usuário entre nomes compostos, apenas o primeiro nome deverá ser considerado, se iniciar com a letra sorteada na rodada.
13. A pontuação de cada resposta equivale à quantidade de caracteres da resposta.
14. Caso dois ou mais jogadores entrem a mesma resposta, ignorando maiúsculas e minúsculas, cada jogador terá como pontuação a fração da quantidade de caracteres correspondente à quantidade de usuários com respostas iguais, com arredondamento. Exemplo: Se quatro jogadores tiverem respondido "Araraquara", cada um fará jus a 10/4 = 2,5 = 3 pontos (arredondando).
15. Ao final de cada rodada, o programa deverá exibir as respostas de todos os jogadores e, em seguida, uma tabela de pontuação parcial (caso ainda haja rodada a ser jogada) ou final (caso seja o fim do jogo).
16. Caso seja o fim do jogo, o programa deverá indicar o jogador ganhador.
17. Caso haja empate em pontos de dois ou mais jogadores, o ganhador será aquele que respondeu todas as rodadas no menor tempo, considerando a soma dos tempos de todas as rodadas.

EXEMPLO

Considere como exemplo o jogo abaixo: (em vermelho, as entradas dos usuários)

```
*** JOGO AMEDONHA ***

Quantos jogadores? 3
Qual é o nome do jogador 1? Antônio
Qual é o nome do jogador 2? Francisco
Qual é o nome do jogador 3? José

A letra desta rodada é: B
A categoria desta rodada é: Nome de Pessoa
A ordem desta jogada será:
  1. Francisco
  2. José
  2. Antônio

Tecle [Enter] para iniciar a rodada: <Enter>
<<< A tela é limpa >>>
Francisco, você deve entrar um "Nome de Pessoa" com a letra "B" em 14 segundos:
Bernardo
```

```
<<< A tela é limpa >>>
José, você deve entrar um “Nome de Pessoa” com a letra “B” em 12 segundos: Bitu

<<< A tela é limpa >>>
Antônio, você deve entrar um “Nome de Pessoa” com a letra “B” em 10 segundos: Beatriz

<<< A tela é limpa >>>


Jogadas realizadas:
Francisco:     Bernardo
José:          Bitu
Antônio:       Beatriz


Concluída a rodada, esta é a tabela de escores:

             Nome
             de       Total
             Pessoa   Parcial

Antônio.      7        7
Francisco     8        8
José.         4        4

Tecle [Enter] para iniciar a próxima rodada: <Enter>
<<< A tela é limpa >>>

A letra desta rodada é: F
A categoria desta rodada é: Nome de Comida
A ordem desta jogada será:
  1. José
  2. Antônio
  2. Francisco

Tecle [Enter] para iniciar a rodada: <Enter>

<<< A tela é limpa >>>
José, você deve entrar um “Nome de Comida” com a letra “F” em 14 segundos: Farofa

<<< A tela é limpa >>>
Antônio, você deve entrar um “Nome de Comida” com a letra “F” em 12 segundos:
Feijoada

<<< A tela é limpa >>>
Francisco, você deve entrar um “Nome de Comida” com a letra “F” em 10 segundos:
Feijoada

<<< A tela é limpa >>>

Jogadas realizadas:
José:          Farofa
Antônio:       Feijoada
Francisco:     Feijoada

Concluída a rodada, esta é a tabela de escores:

             Nome      Nome
             de        de        Total
             Pessoa    Comida    Parcial

Antônio.      7         4         11
Francisco     8         4         12
José          4         6         10

<<< As demais rodadas seguem da mesma forma >>>
```

```
RESULTADO FINAL:

             Nome.     Nome      Nome      Nome
             de        de        de        de        Total
             Pessoa    Comida    Animais   Cidade    Geral

Antônio       7         4         6         8         25
Francisco     8         4         0         5         17
José          4         6         8         5         23

O ganhador é: Antônio
```

REQUISITOS NÃO FUNCIONAIS

1. O programa DEVE utilizar o mínimo de memória possível.
2. O programa DEVE utilizar gerenciamento dinâmico de memória.
3. O projeto deverá ser constituído por códigos fontes separados, de acordo com características funcionais (Ex. estrutura de dados, programa principal, funções auxiliares, etc.). Todos os arquivos-fonte com o respectivo Makefile para construir o projeto devem ser disponibilizados na entrega.
4. O dimensionamento das variáveis a serem utilizadas deve otimizar a ocupação de memória.

Critérios de Avaliação

A avaliação será realizada em três fases:

1. Análise do código-fonte;
2. Análise da execução do programa (teste);
3. Apresentação do protótipo pelos membros da equipe.

O código-fonte será avaliado de acordo com os seguintes critérios qualitativos:

i. Eficácia do programa em suprir todos os requisitos funcionais e não funcionais;
ii. Eficiência do programa (otimização);
iii. Organização do código (uso racional de subprogramas, estruturas, etc.);
iv. Legibilidade do código (uso de endentação e semântica dos identificadores de variáveis);
v. Documentação (comentários dentro do código fonte).

Obviamente, funcionalidades adicionais às que foram solicitadas neste documento são bem-vindas e serão gratificadas na nota (na medida do possível). O código-fonte deve conter, em comentário no início, os nomes e matrículas dos alunos que compõem o grupo, assim como uma descrição sucinta da atuação de cada um no desenvolvimento do projeto. O código-fonte deve ser submetido na data fixada através do SIGAA. A apresentação do protótipo, se necessária, será marcada em seguida.

Lembramos que todos os programas serão submetidos a análise léxica automática, que pode evidenciar cópia de código (plágio).

A apresentação do protótipo deverá ser gravada pelos membros da equipe (sugestão: com o Google Classroom), com duração de até 10 minutos. Cada um dos membros da equipe deverá mostrar a parte do projeto de sua responsabilidade e deverá demonstrar domínio sobre o projeto como um todo. As notas dos membros de uma equipe podem ser diferentes, dependendo do desempenho dos membros na apresentação, a critério do professor avaliador.

Os trabalhos serão corrigidos no **Linux**. Portanto, certifique-se que o trabalho feito no Windows também compila e roda no Linux.