

---

**DEPARTAMENTO DE SISTEMAS E COMPUTAÇÃO**  
**Faculdade de Engenharia - UERJ**  
**Laboratório de Programação I (FEN06-04049)**

Tempo: 3 horas

Prova Final - 2024-2

16 de janeiro de 2025

Prof. João Araujo

---

1. Em 1953, com a descoberta do DNA, por James Watson e Francis Crick comprovaram-se as teorias de Charles Darwin sobre a origem da espécie. Sabendo que uma base A se liga com uma base T e uma base G com uma base C, faça um programa que, dada uma linha com a sequência de bases de DNA, gere o outro lado da espiral de DNA. **(2pts)**
2. Chapeuzinho Vermelho estava cansada de ter sua conta invadida por hackers, veio pedir sua ajuda para criar um programa que a ajudasse a escolher uma boa senha. Ela pediu que o programa tivesse duas formas de atuar: interativa, indicado pelo argumento da linha de comando “-i”; e validação direta, indicada pelo argumento “-v”. Vejamos as condições:

No modo de validação imediata -v (**item a=1pt**), o programa valida uma string de entrada logo após o argumento. Assim, o programa poderia ser chamado das seguintes formas:

*valida\_senha -v banana*  
*valida\_senha -i*

E o programa escreveria aceita ou rejeitada para a senha escolhida. No caso acima, a senha seria rejeitada (veja abaixo o porquê).

No modo interativo -i (*valida\_senha -i*) (**b=1pt**), quando ela aceita várias linhas do usuário, cada uma com uma senha e valida esta senha. Cada senha e resultado devem ser guardados em um lista encadeada (**c=1pt**) que será despejada na tela assim que o usuário digitar uma linha vazia (**d=1pt**).

Sem saber o que seria uma boa senha, você consultou um hacker profissional, chamado **L.o30.M4u**, que te deu dicas do que seria uma senha segura:

- Senhas devem ser constituídas por uma mistura de letras minúsculas, maiúsculas e números somente, com ao menos um de cada. (**e=2pts**)
- Devem ter mais de 5 caracteres. (**f=1pt**)

Alguns exemplos:

**Brasiluerj** rejeitado por conter apenas letras.

**a3r4** rejeitada por ser muito curta.

**ahh5k6t** rejeitada por não conter letra maiúscula.

O programa também deve emitir mensagem de erro se for pedida uma opção inválida (**g=1pt**). A lista encadeada deve ser constituída de estruturas que armazenam a senha e o resultado da análise.

Programas que dividem o problema em vários problemas menores têm mais chance de dar certo.

O Programa deve ter pelo menos 3 funções que serão chamadas pelo **main()** (penalidade por não seguir essa regra -0.5pt).

Vida Longa e Próspera, mas cada variável global desconta um ponto...

Envie os arquivos provap1-q{12}-nomeAluno.c, sem o executável, para o classroom.

Obs.: Qualquer tentativa de **desonestidade intelectual**, conhecida popularmente como cola, será considerada falta grave e punida com nota **zero** na prova. Provas com **erros de sintaxe** (ou seja, erros de compilação) receberão nota **ZERO**. Seu programa deve ser compilável sem erros. Evite também warnings, pois eles podem indicar na realidade erros lógicos de seu programa.

Seu programa deve estar bem indentado, documentado e organizado. A indentação deve deixar clara a estrutura de subordinação dos comandos. Os comentários devem ser esclarecedores.

---