



Avaliação

Nome:

Turma: TSI 1AN

Data: 20/07/2016

Observações:

- A avaliação deverá ser resolvida nos computadores da instituição.
- Não poderá haver nenhum tipo de consulta, salvo o material permitido pelo professor.
- A resolução da avaliação deverá ser entregue para o professor por meio digital, através do dispositivo de armazenamento disponibilizado pelo mesmo para a cópia.
- Os arquivos que compõe a resolução deverão estar dentro de uma pasta com o nome do aluno, e os nomes dos arquivos deverão conter o número da questão.

- 1) Escreva uma função recursiva que ajude o usuário com sua dívida. A função deve receber como parâmetro a quantia ele recebeu de empréstimo, a taxa de juros mensais e quanto tempo ele irá levar para pagar, e retornar a quantia que deverá ser paga ao fim do tempo transcorrido. A solução iterativa se encontra ao lado. **(25%)**

```
int q=5000; //$$$  
int t=12; //tempo 12 meses  
int j=0.03; //juros 3%  
int i;  
for (i=0 ; i<t ; i++)  
    q += q*j;
```

- 2) Supondo um arquivo de texto chamado nomes.txt e outro snomes.txt. Ambos contém strings de no máximo 20 caracteres em cada linha. O número máximo de linhas é 1000. Escreva um programa que leia e armazene em uma matriz de nomes o conteúdo do arquivo nomes.txt, e uma matriz de sobrenomes o conteúdo do arquivo snomes.txt. Caso algum dos arquivos não exista, o programa deve mostrar uma mensagem de erro e encerrar. Então peça para o usuário informar em N quantos nomes completos deseja gerar. Mostre na tela N nomes completos gerados a partir de uma string aleatória da matriz nome seguido de duas strings aleatórias da matriz de sobrenomes. Ao fim, pergunte para o usuário se ele deseja salvar os nomes, e em caso afirmativo peça para ele informar o nome do arquivo e concatene o nome informado com a extensão ".txt" e grave os N nomes gerados no arquivo escolhido pelo usuário, um em cada linha. **(35%)**
- 3) Crie uma struct para guardar o resultado de um bolão de 10 jogos. A struct deve guardar o nome do jogador, telefone, e as apostas dos resultados dos 10 jogos. Cada aposta consiste em dizer qual dos dois times irá ganhar, ou se dará empate ('1', '2' ou 'e'). Crie funções para guardar e carregar automaticamente o conteúdo da struct em um arquivo externo ao fim e início do programa (permissão binária). Crie um programa que ofereça as opções de sair, inserir jogador e verificar ganhador. Caso o jogador escolha cadastrar, peça os dados e as apostas do jogador, e cadastre este jogador em um vetor struct. O vetor struct deverá ser alocado dinamicamente. Caso o jogador escolha verificar vencedor, chame uma função que peça o resultado de cada um dos 10 jogos. Então a função deve contar os acertos que cada jogador conseguiu, e retornar o código do jogador vencedor (ignore empate). Mostre no main o nome do jogador vencedor e telefone. **(40%)**