

Acadêmico(a):		
Curso:	Engenharia de Software	Matrícula:
Disciplina:	FGA0147 - Estruturas de Dados e Algoritmos	Período para Realização:
Professor:	Dr. Nilton Correia da Silva	01/12/2022. Das 10h às 11h50min
Orientações Gerais 1. DESLIGUE (ou use o modo silencioso) E GUARDE O CELULAR. 2. Preencha seu nome e número de matrícula. 3. A interpretação das questões é parte do processo de avaliação, não sendo permitidas consultas ou comunicação entre alunos. 4. Esta avaliação deve ser preenchida à caneta de tinta cor azul ou preta.		NOTA (0-10):

PROVA TEÓRICA 1 – PT1

1. [1,5] O que será impresso no programa do **quadro Ponteiro**?

Resposta: _____

2. [1,5] O programa do **quadro Nomes** deve ler o nome completo de uma pessoa e imprimir apenas os sobrenomes. **Complete a linha 11** do código para que a função *acheSobrenome* fique correta.

Resposta: _____

3. [2,0] Considerando que as funções do **quadro Média** devem retornar a média dos valores de um vetor do tipo *float*, **marque a opção contendo a afirmativa correta.**

- ☐ – *Media* e *Media2* estão corretas somente no caso de *pV* ter sido alocado dinamicamente;
- ☐ – *Media* e *Media2* estão corretas somente no caso de *pV* não ter sido alocado dinamicamente;
- ☐ – *Media* está correta somente no caso de *pV* não ter sido alocado dinamicamente;
- ☐ – *Media2* está correta somente no caso de *pV* ter sido alocado dinamicamente;
- ☐ – *Media* e *Media2* estão corretas, sendo *pV* um vetor declarado estaticamente ou alocado dinamicamente.

4. [2,5] Considere o problema de se contar quantas vezes um dígito ocorre em um número natural e marque a opção correta, conforme o **Quadro Contagem**.

- ☐ – Apenas a função *Repete4* está correta
- ☐ – As funções *Repete2* e *Repete4* estão corretas
- ☐ – As funções *Repete1* e *Repete3* estão corretas
- ☐ – As funções *Repete1*, *Repete2* e *Repete3* estão corretas
- ☐ – Todas as soluções estão corretas

5. [2,5] A função de Ackermann é definida para valores inteiros e não negativos m e n , conforme abaixo. Marque a opção abaixo que contém o resultado correto para $A(2,3)$.

$$A(m, n) = \begin{cases} n + 1, & \text{se } m = 0 \\ A(m - 1, 1), & \text{se } m > 0 \text{ e } n = 0 \\ A(m - 1, A(m, n - 1)), & \text{se } m > 0 \text{ e } n > 0 \end{cases}$$

- a. [] – $A(2,3) = 40$
- b. [] – $A(2,3) = 9$
- c. [] – $A(2,3) = 6$
- d. [] – $A(2,3) = 4$
- e. [] – $A(2,3) = 14$

Tenha uma excelente avaliação...

Quadro Ponteiro: Código sobre ponteiro para vetor de inteiro.

```
1 #include <stdio.h>
2 int main (void)
3 {
4     int vet[] = {10,20,30};
5     int *p = vet;
6     p++;
7     printf("\n%d", *p);
8     (*p)++;
9     printf(" - %d", *p);
10    printf(" - %d", *(p+1));
11    return 0;
12 }
```

Quadro Nomes: Função para extrair sobrenomes.

```
1 #include <stdio.h>
2 char * acheSobrenome (char nome [])
3 {
4     char * pnome ;
5     int i = 0;
6     while ( nome [ i ] != ' ')
7     {
8         i++;
9     }
10    i++;
11    pnome = _____ //COMPLETE O CÓDIGO AQUI.
12    return pnome;
13 }
14
15 int main (void)
16 {
17     char nomeCompleto [80];
18     char *p ;
19     puts ( "Entre com o seu nome e um sobrenome:" );
20     gets (nomeCompleto);
21     p = acheSobrenome (nomeCompleto);
22     puts ( p );
23     return 0;
24 }
```

Quadro Média: Conjunto de códigos sobre média de um vetor numérico.	
<pre>float Media(float *pV, int pn) { float pM = 0; int pInd; for(pInd=0; pInd<pn; pInd++) pM += pV[pInd]; return (pM/pn); }</pre>	<pre>float Media2(float *pV, int pn) { float *pVaux = pV, pM = 0; int pInd; for(pInd=0; pInd<pn; pInd++) { pM += *pVaux; pVaux++; } return (pM/pn); }</pre>

Quadro Contagem: Conjunto de códigos sobre contagem de dígitos.	
<pre>int Repete1(int pnumero, int pdigito) { int pdig, pdiv, pcont = 0; if(pnumero < 10) { if(pnumero == pdigito) return 1; else return 0; } pdiv = pnumero/10; pdig = (pnumero - pdiv*10); if(pdig == pdigito) pcont = 1; return pcont + Repete1(pdiv,pdigito); }</pre>	<pre>int Repete2(int pnumero, int pdigito) { int pdig, pdiv; if(pnumero < 10) { if(pnumero == pdigito) return 1; else return 0; } pdiv = pnumero/10; pdig = (pnumero - pdiv*10); if(pdig == pdigito) return Repete2(pdiv,pdigito+1); else return Repete2(pdiv,pdigito); }</pre>
<pre>int Repete3(int pnumero, int pdigito, int pqtde) { int pdig, pdiv; if(pnumero < 10) { if(pnumero == pdigito) return pqtde + 1; else return pqtde; } else { pdiv = pnumero/10; pdig = (pnumero - pdiv*10); if(pdig == pdigito) return Repete3(pdiv, pdigito, pqtde+1); else return Repete3(pdiv, pdigito, pqtde); } }</pre>	<pre>int Repete4(int pnumero, int pdigito, int pqtde) { int pdig, pdiv; if(pnumero < 10) { if(pnumero == pdigito) return 1; else return 0; } else { pdiv = pnumero/10; pdig = (pnumero - pdiv*10); if(pdig == pdigito) return Repete4(pdiv, pdigito, pqtde+1); else return Repete4(pdiv, pdigito, pqtde); } }</pre>