

Estrutura de Dados - 3º Período SI / TADS

Prof. André Cypriano Monteiro Costa

Lista de Revisão - Ponteiros

1 O que á um nente	iro2 Dara que conve un	m nontoiro?					
1. O que é um ponteiro? Para que serve um ponteiro?							
2. Quais das seguinte	es instruções são corre	etas para declarar um	ponteiro?				
a) int _ptr x;	b) int *ptr;	c) *int ptr	d) *x;				
3. Qual é a maneira	correta de referencia	r o conteúdo de ch , a	ssumindo que	o endereço de ch já			
foi atribuído ao pont	eiro indica ?						
a) *indica;	b) int *indica;	c) *indic;	d) ch;	e) *ch;			
4. Na expressão floa	t *pont ; o que é do tip	oo float?					
a) a variável pont.		b) o endereço de pont.					
c) a variável apontada por pont.		d) nenhuma das anteriores.					
5. Assumindo que o	o endereço de num	foi atribuído ao por	nteiro pnum , o	quais das seguintes			
expressões são verda	adeiras?						
a) num == &pnum		b) num == *pnum;					
c) pnum == *num;		d) pnum == #					
6. Seja o seguinte tre	echo de programa:						
int i=3, j=5;							
int *p, *q;							
p = &i							
q = &j							
Qual é o valor das se	guintes expressões?						
a) p == &i	b) *p - *q	c) **&p	d) 3 - *p / (*q) + 7			

7. Se i e j são variá	veis inteiras e p e	q ponteiros para int,	quais das seguintes ex	xpressões de
atribuição são ilegais	?			
a) p = &i	b) *q = &j	c) p = &*&i	d) i = (*&)j;	e) i = *&j
f) i = *&*&j	g) q = *p;	h) i = (*p)++ + *q;		
8. Supor a declaração	o: int mat[4], *p, x;	Quais expressões são v	válidas? Justifique:	
a) p = mat + 1;		b) p = mat++;		
c) p = ++mat;		d) x = (*mat)++;		
0.0 1/2 2/12 12				
9. Qual é a saída do p	orograma a seguir?			
#include <stdio.h></stdio.h>				
int main()				
{ int yet[] (4)	n 12).			
int vet[] = {4,9	9,13};			
int i;	.)			
for(i=0;i<3;i++	("%d ",*(vet+i));			
for(i=0;i<3;i++	/*) /"%X ",vet+i);			
	(/0X ,VE(+I),			
return 0;				
}				
10. Seja a seguinte se	equência de instruç	cões em um programa C	: :	
int *pti;				
int i = 10;				
pti = &i				
Qual é a afirmativa fa	alsa?			
a) pti armazena o en	dereço de i			
b) *pti é igual a 10				
c) ao se executar *pt	i = 20, i passará a to	er o valor 20		
d) ao se alterara o va	ılor de i, *pti será n	nodificado		
e) pti é igual a 10				

11. Seja vet um vetor de 4 elementos: **TIPO vet[4]**. Supor que depois da declaração, vet esteja armazenado no endereço de memória 4092 (ou seja, o endereço de vet[0]). Supor também que na máquina usada uma variável do tipo char ocupa 1 byte, do tipo int ocupa 2 bytes, do tipo float ocupa 4 bytes e do tipo double ocupa 8 bytes. Qual o valor de vet+1, vet+2 e vet+3 se:

a) vet for declarado como char?

b) vet for declarado como int?

c) vet for declarado como float?

d) vet for declarado como double?

12. Responda as perguntas abaixo. Se achar necessário, faça testes no programa em C.

```
a) Explique a diferença entre p++; (*p)++; *(p++);
```

- b) O que quer dizer *(p+10);?
- 13. Qual o valor de y no final do programa? Tente primeiro descobrir e depois verifique no computador o resultado. A seguir, escreva um /* comentário */ em cada comando de atribuição explicando o que ele faz e o valor da variável à esquerda do '=' após sua execução.

```
#include<stdio.h>
int main()
{
    int y, *p, x;
    y = 0;
    p = &y;
    x = *p;
    x = 4;
    (*p)++;
    x--;
    (*p) += x;
    printf ("y = %d\n", y);
    return(0);
}
```

14. Diga quais expressoes abaixo sao válidas ou não. Considere as declarações

```
int vetor[10];
int *ponteiro;
```

```
a) vetor = vetor + 2;
b) vetor++;
c) vetor = ponteiro;
d) ponteiro = vetor;
e) ponteiro = vetor+2;
15. Dado o programa abaixo, completar as Tabelas 1 e 2:
#include<stdio.h>
int main()
{
       int i, j, *p_1, *p_2, **p_p_1, **p_p_2;
       i = 4;
       j = 5;
       p_1 = &i;
       p_2 = k_j;
       p_p_1 = &p_2;
       p_p_2 = &p_1;
}
```

Tabela 1

Nome Variável	i	j	p_1	p_2	p_p_1	p_p_2
Conteúdo	4	5				
Endereço	1000	1007	1030	1053	1071	1079

Tabela 2

Expressão	i	*p_2	&i	&p_2	**p_p_1	*p_p_2	&*p_1	j	*p_1	&p_1
Resultado										