

Lista de exercícios

Assunto: ponteiros e alocação dinâmica

1. Considerando o trecho de código e o mapa de memória abaixo, defina o valor de retorno de cada expressão:

```
char i = 605, j = 5;
char *p, *q;
p = &i;
q = &j;
```

Endereço	Variável	Conteúdo
500	i	650
...		
650	j	5

- a) $p == \&j$
 b) $p \leq q$
 c) $*p + *q * i$
 d) $*q + (\&i)$
 e) $5 + *(\&j)$
2. Dada a seguinte situação, defina o valor de retorno de cada expressão. Considere que p é um ponteiro para char (char *p):

		p																	
		└─																	
Endereços:	...	345	346	347	348	349	350	351	352	...									
Conteúdo:	...	6	8	3	9	9	1	1	10	memória não alocada									

- a) $*p$
 b) $*(p + 5)$
 c) $*p + 5$
 d) $p[2] + *(p + 3)$
 e) $p + 10$
 f) $*p + 10$
 g) $(p + 6) + *p$
 h) $*((p + 6) - 5) * *(p + 7)$

Implementação:

Os exercícios abaixo devem ser feitos usando somente ponteiros, sem usar índices de vetor (`vet[i]`, etc).

Os vetores utilizados devem ser alocados dinamicamente.

3. Escreva um programa que leia N inteiros da entrada padrão, armazene-os em um vetor e os escreva na saída padrão na ordem contrária a de leitura, tal como exibido no exemplo abaixo.

entrada:

$v1 = \{10, 20, 30\}$

saída:

$v1 = \{30, 20, 10\}$

4. Escreva um programa para calcular o tamanho de uma string usando somente ponteiros.
5. Crie uma função que irá receber dois vetores de inteiros, $v1$ e $v2$, e seus respectivos tamanhos, $n1$ e $n2$. Essa função deverá retornar um ponteiro para um terceiro vetor ($v3$), o qual deverá armazenar a união de $v1$ e $v2$. Dessa forma, o tamanho de $v3$ será definido como $n1 + n2$. Utilize uma variável passada por referência para armazenar o tamanho do vetor $v3$. Exiba o conteúdo do vetor $v3$ na função principal.

Exemplo:

Entrada:

$v1 = \{11, 13, 45, 7\}$

$v2 = \{24, 4, 16, 81, 10, 12\}$

Saída:

$v3 = \{11, 13, 45, 7, 24, 4, 16, 81, 10, 12\}$