Universidade Federal da Bahia

Instituto de Matemática

Departamento de Ciência da Computação

Disciplina: Estrutura de Dados I

Aluno: Pedro Gabriel Carrano

**1- Resposta**

A diferença conceitual é que a primeira expressão (v[3]) é o valor em si enquanto que a segunda (v + 3) é um ponteiro para ele.

|  |
| --- |
| **2 – Resposta** |
|  |
| a) |
|  |
| Quando é colocado p=x ele está apontando para o endereço do valor 100. Para imprimir o valor da variável x, o certo seria p=&x.  #include <stdio.h>  #include <stdlib.h>  int main(){  int x,\*p;  x=100;  p=&x;  printf("Valor de p: %d.\n",\*p);\\ poderíamos imprimir o valor do endereço usando apenas p  } |
|  |
| b) O certo seria declarar ‘int temp’ e o usar como uma variável (sem usar \*temp). |
|  |
|  |
|  |
| c) Poderia ser usar ‘strcmp’ e analizado se o resultado é positivo ou negativo. |
|  |
| **3 - Resposta**  Se o endereco de v[0] é 55000 então v = 55000. v + 3 estamos adicionando 3 bytes. Ficaria v = 55024.  **4- Resposta**  **5- Resposta**  Resultado é de 1 até 98 depois até 1 novamente. Os elementos são sobrescritos ao executar o for.  **6- Resposta**  void troca(float\* a, float\* b) {  float t;  \*a=t;  \*a=\*b;  \*b=t;  }  int main() {  float a=0, b=1;  troca(&a, &b);  printf("A=%.0f\nB=%.0f\n", a,b);  }  **7- Resposta**  **8- Resposta**  **9- Resposta**  int x = 100, \*p, \*\*pp; \\ x recebe valor 100, ponteiro p e ponteiro para ponteiro pp  p = &x; \\ p aponta para o endereço do valor de x  pp = &p; \\ pp aponta para o endereço do ponteiro p  printf(“Valor de pp: %d\n”, \*\*pp); \\ é impresso o valor de x = 100  **10- Resposta** |