1)

 A criação de um ponteiro só faz sentido se for associado a algum endereço de memória, para isso usa-se a seguinte sintaxe:  
1. int idade = 18;  
2. int \*ponteiro\_para\_idade = &idade;

Na linha 1 criamos uma variável primitiva inteira com valor 18 e na linha 2 associamos um ponteiro chamado ponteiro\_para\_idade ao endereço da variável primitiva idade.

Podemos imprimir o conteúdo do ponteiro, que será o endereço da variável que ele aponta. Utilizando o ponteiro criado anteriormente (ponteiro\_para\_idade) temos a seguinte sintaxe:

**Alternativas:**

* a)

printf("\n Conteudo do ponteiro: %d", &idade);

* b)

scanf("\n Conteudo do ponteiro: %p", ponteiro\_para\_idade);

* c)

gets("\n Conteudo do ponteiro: %d", &idade);

* d)

printf("\n Conteudo do ponteiro: %p", ponteiro\_para\_idade);

Alternativa assinalada

* e)

scanf("\n Conteudo do ponteiro: %d", &idade);

2)

As variáveis são muito úteis para o armazenamento de diversos tipos de dados.

De acordo com as informações apresentadas na tabela a seguir, faça a associação dos tipos de variáveis com suas respectivos características.

|  |  |  |
| --- | --- | --- |
| Coluna -A |  | Coluna -B |
| I. Vetor |  | 1. Podemos manipular variáveis e outros recursos pelo endereço de memória. |
| II. Matriz |  | 2. Possuem a estrutura de uma tabela contendo apenas 1 coluna e N linhas. |
| III. Struct |  | 3. Possuem a estrutura de uma tabela contendo apenas N coluna e N linhas. |
| IV. Ponteiro |  | 4. Um tipo de variável composta heterogênea. |

I - 2; II - 3; III - 4; VI - 1.

Assinale a alternativa que apresenta a associação correta.

**Alternativas:**

* a)

I - 3; II - 2; III - 1; IV- 4.

* b)

I - 2; II - 3; III - 1; IV - 4.

* c)

I - 4; II - 1; III - 2; IV - 3.

* d)

I - 1; II - 2; III - 3; IV - 4.

* e)

I - 2; II - 3; III - 4; IV - 1.

Alternativa assinalada

3)

 Manzano, Matos e Lourenço(2015) afirmam que a criação de um vetor é similar a uma variável primitiva tendo que acrescentar apenas um número entre colchetes indicando qual será o tamanho desse vetor por exemplo: *int valores [15]*, neste caso, estamos criando 15 espaços para armazenar valores inteiros. Cada elemento no vetor é acessado através do seu índice, que sempre começará pelo valor zero, independentemente da linguagem de programação.

Observe o programa a seguir utilizando um vetor.

*#include <stdio.h>*

*const int valor = 3;*

*int main ()*

*{*

*int vetor[4]={2,4,6,8};*

*vetor [0] = vetor [0] \* valor;*

*vetor [1] = vetor [1] \* valor;*

*vetor [2] = vetor [2] \* valor;*

*vetor [3] = vetor [3] \* valor;*

*printf ("\n  %d - ", vetor [3]);*

*printf ("\n  %d -", vetor [2]);*

*printf ("\n  %d -", vetor [1]);*

*printf ("\n  %d .", vetor [0]);*

*return 0;*

*}*

Assinale a opção correta que apresenta o resultado que será impresso na tela após o programa ser executado.

**Alternativas:**

* a)

6 – 8 – 10 – 12.

* b)

24 – 18 – 12 – 6.

Alternativa assinalada

* c)

6 – 12 – 18 – 24.

* d)

18 – 12 – 6 – 24.

* e)

24 – 12 – 18 – 6.

4)

Conforme Mizrahi (2008) um vetor é uma estrutura de dados do mesmo tipo primitivo. Possui um índice que deve ser rigorosamente respeitado, não podemos por exemplo armazenar mais valores do que a quantidade que foi informada na declaração do vetor.

Observe o programa a seguir que realiza a troca de elementos entre o próprio vetor.

|  |
| --- |
| *#include <stdio.h>*  *const int TAM = 2;*  *int main ()*  *{*  *int idade[TAM]={25,48};*  *int troca;*  *printf ("\n Antes da Troca: ");*  *printf ("\n  %d ", idade [0]);*  *printf ("\n  %d ", idade [1]);*  *troca = idade[0];*  *idade [0]= idade [1];*  *idade [1] = troca;*  *printf ("\n Depois da Troca: ");*  *printf ("\n  %d ", idade [0]);*  *printf ("\n  %d ", idade [1]);*  *return 0;*  *}* |

Fonte: MIZRAHI, V. V. Treinamento em linguagem C. 2ª ed. São Paulo. Pearson Prentice Hall, 2008.

Com base nas afirmações sobre vetor e o programa apresentado, avalie as seguintes asserções e a relação proposta entre elas:

I. Existem algumas facilidades que podem ser utilizadas para ajudar o trabalho do programador e é o caso da utilização de constantes. Podemos utilizar uma constante para determinar o tamanho do vetor. Primeiro criamos uma constante do tipo inteiro e atribuímos um valor. Ao criar o vetor, usamos a constante para informar o tamanho do vetor.

PORQUE

II. Essa ação pode agilizar o trabalho do programador, suponha que o vetor tenha o tamanho de 30 elementos, para testar o programa devemos informar 30 elementos toda vez. Por exemplo: usando uma constante com o valor 4, basta informar o valor de 4 elementos e o podemos testar o programa com 4 elementos do vetor. No final dos testes, basta modificar o valor 4 da constante para o valor 30 (e que era a proposta inicial do programa).

Analise atentamente as asserções e assinale a alternativa CORRETA.

**Alternativas:**

* a)

As asserções I e II são proposições verdadeiras, e a II é uma justificativa da I.

Alternativa assinalada

* b)

As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa da I.

* c)

A asserção I é uma proposição verdadeira, e a II é uma proposição falsa.

* d)

A asserção I é uma proposição falsa, e a II é uma proposição verdadeira.

* e)

As asserções I e II são proposições falsas.

5)

A programação é utilizada para ajudar a resolver problemas de todos os níveis. Operações matemáticas são os exemplos preferidos para ajudar no entendimento das estruturas das linguagens de programação.  A área de um quadrado corresponde ao tamanho da superfície desta figura. Um quadrado é um quadrilátero que possui seus lados congruentes ou seja, eles possuem exatamente a mesma medida.

O programa a seguir que calcula a área de um quadrado, observe que está faltando uma linha no programa:

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10. | *#include <stdio.h>*  *int lado, area;*  *int main()*  *{*  *printf("Insira o Lado: ");*  *scanf("%d", &lado);*  *// Cálculo da área*  *printf("A área e: %d \n", area);*  *return 0;*  *}* |

Assinale a opção correta que apresenta o comando que deve ser inserido na linha 7, para calcular a área do quadrado:

**Alternativas:**

* a)

*area = (lado1 \* lado2);*

* b)

*area = (lado \* 2);*

* c)

*area = (lado)2;*

* d)

*area = (lado \* lado);*

Alternativa assinalada

* e)

*lado = (area \* area);*