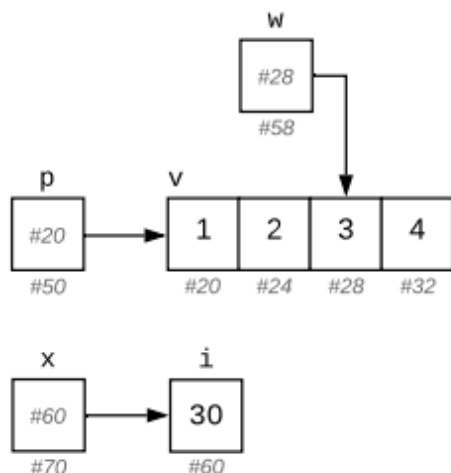


Relação entre ponteiros e vetores



* A variável *i* armazena um *int*
 * O vetor *v* também é armazena valores do tipo *int*

Escreva um trecho de código que reproduza a ilustração acima.

```
#include <stdio.h>

int main()
{
    int i = 30;
    int v[4] = {1, 2, 3, 4};
    int* p = v;
    int* w = &v[2];
    int* x = &i;

    return 0;
}
```

Determine os valores com base na ilustração

p: <u>#20</u>	p + 3: <u>#32</u>
v: <u>#20</u>	&p[3]: <u>#32</u>
&v[0]: <u>#20</u>	*(p+3): <u>4</u>
v + 1: <u>#24</u>	p[3]: <u>4</u>
*(v+2): <u>3</u>	
w: <u>#28</u>	
*w: <u>3</u>	
w[1]: <u>4</u>	
x: <u>#60</u>	
&x: <u>#70</u>	
*x: <u>30</u>	
x[0]: <u>30</u>	

Escreva um trecho de código que imprima os valores 1, 2, 3 e 4 a partir da variável *p*. Utilize a notação de sua preferência.

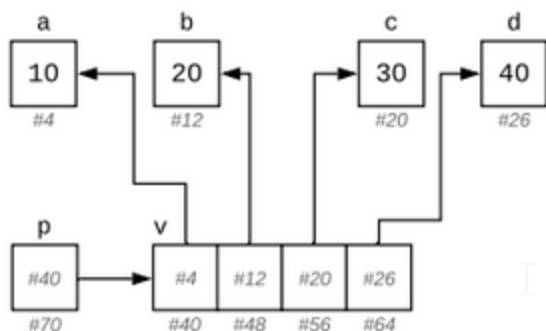
```
#include <stdio.h>

int main()
{
    int i = 30;
    int v[4] = {1, 2, 3, 4};
    int* p = v;
    int* w = &v[2];
    int* x = &i;

    printf("%p \n", &p[3]);
    printf("%p \n", p + 3);

    return 0;
}
```

Vetor de Ponteiros



* As variáveis a, b, c, d armazenam valores do tipo int

Determine os valores com base na ilustração

p: #40
*p: #4
**p: 10
p[1]: #12
*p[1]: 20
**(p+2): 30
v: #40
v[3]: #26
v+3: #56
*(v+3): #26
**(v+3): 40

Escreva um trecho de código que reproduza a ilustração acima.

```
#include <stdio.h>

int main()
{
    int a = 10;
    int b = 20;
    int c = 30;
    int d = 40;

    int* v[4] = {&a, &b, &c, &d};
    int** p = v;

    return 0;
}
```

Escreva um trecho de código que imprima os valores 10, 20, 30 e 40. Não é permitido o uso de colchetes. Utilize a notação de ponteiros (* e aritmética de ponteiro)

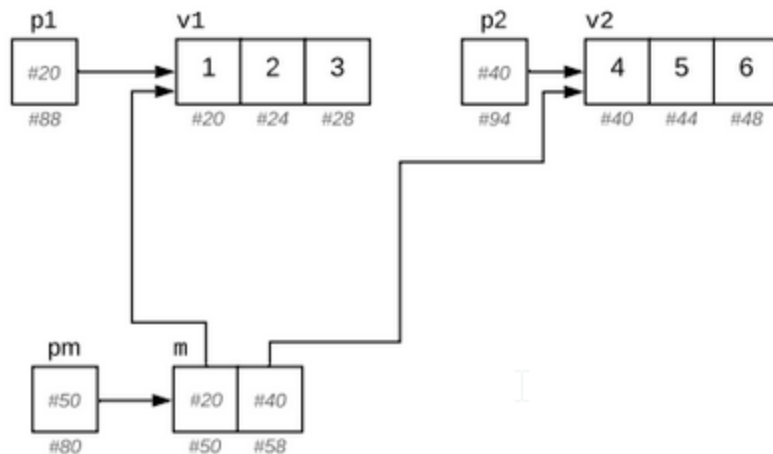
10

```
int a = 10;
int b = 20;
int c = 30;
int d = 40;

int* v[4] = {&a, &b, &c, &d};
int** p = v;
```

```
for (int i = 0; i < 4; i++)
{
    printf("%d \n", **(p + i));
}
```

Vetor de Ponteiros



Determine os valores com base na ilustração

p1 : <u>#20</u>	m[0] : <u>#20</u>
v1 : <u>#20</u>	*(m+1) : <u>#40</u>
p1 + 2 : <u>#28</u>	*pm[1] : <u>4</u>
v1 + 2 : <u>#28</u>	**m[1] : <u>4</u>
p2[1] : <u>5</u>	m[0][2] : <u>3</u>
*(v2+1) : <u>5</u>	pm : <u>#80</u>
m : <u>#50</u>	pm+1 : <u>#58</u>
*m : <u>#20</u>	*(pm+1) : <u>#40</u>
pm : <u>#80</u>	*(pm+1) + 2 : <u>#48</u>
*pm : <u>#20</u>	*(*(pm+1) + 2) : <u>6</u>

Escreva um trecho de código que reproduza a ilustração acima.

```
#include <stdio.h>

int main()
{
    int v1[3] = {1, 2, 3};
    int v2[3] = {4, 5, 6};

    int* p1 = v1;
    int* p2 = v2;
    int* m[2] = {v1, v2};

    int** pm = m;

    return 0;
}
```

Escreva um trecho de código que imprima os valores 1, 2, 3, 4, 5 e 6. Utilize a notação de sua preferência.

Não cabe aqui todo o código.
Irei anexar um arquivo com o nome
"File01.c", com o código nele inserido!