

ACH2001 - Introdução à Programação

Matrizes

Prof. Flávio Luiz Coutinho
flcoutinho@usp.br

Matrizes

Equivalente ao conceito matemático de matriz:

- conjunto de elementos organizados em linhas e colunas
- cada elemento descrito por dois índices.

Matrizes são vetores com duas dimensões:

- vetor: `a[i]`
- matriz: `m[lin][col]`

A organização da informação em 2 dimensões pode ser conveniente para representar certos tipos de dados: matrizes (sentido matemático), imagens, listas de listas, etc.

Matrizes

```
int main() {  
  
    int m[3][4]; // matriz com 3 linhas e 4 colunas  
  
    ...  
  
    printf("%d\n", m[0][0]); // 1.a linha e 1.a coluna  
    printf("%d\n", m[0][3]); // 1.a linha e 4.a coluna  
    printf("%d\n", m[2][0]); // 3.a linha e 1.a coluna  
    printf("%d\n", m[2][3]); // 3.a linha e 4.a coluna  
  
}
```

Matrizes

```
int main(){

    int lin, col;

    int m[3][4] = { { 1, 2, 3, 4 }, { 5, 6, 7, 8 }, { 9, 10, 11, 12 } };

    for(lin = 0; lin < 3; lin++){

        for(col = 0; col < 4; col++) printf("%2d ", m[lin][col]);

        printf("\n");

    }

    return 0;

}
```

Matrizes

```
int m[3][4] = { ... };
```

m

14520

18340

18340 (0) 18344 (1) 18348 (2) 18352 (3)



1	2	3	4
5	6	7	8
9	10	11	12

Matrizes

```
int m[3][4] = { ... };
```

```
int * a = (int *) m;
```

m

14520

18340

1	2	3	4	5	6	7	8	9	10	11	12
---	---	---	---	---	---	---	---	---	----	----	----

0

1

2

3

4

5

6

7

8

9

10

11

m[0]

m[1]

m[2]

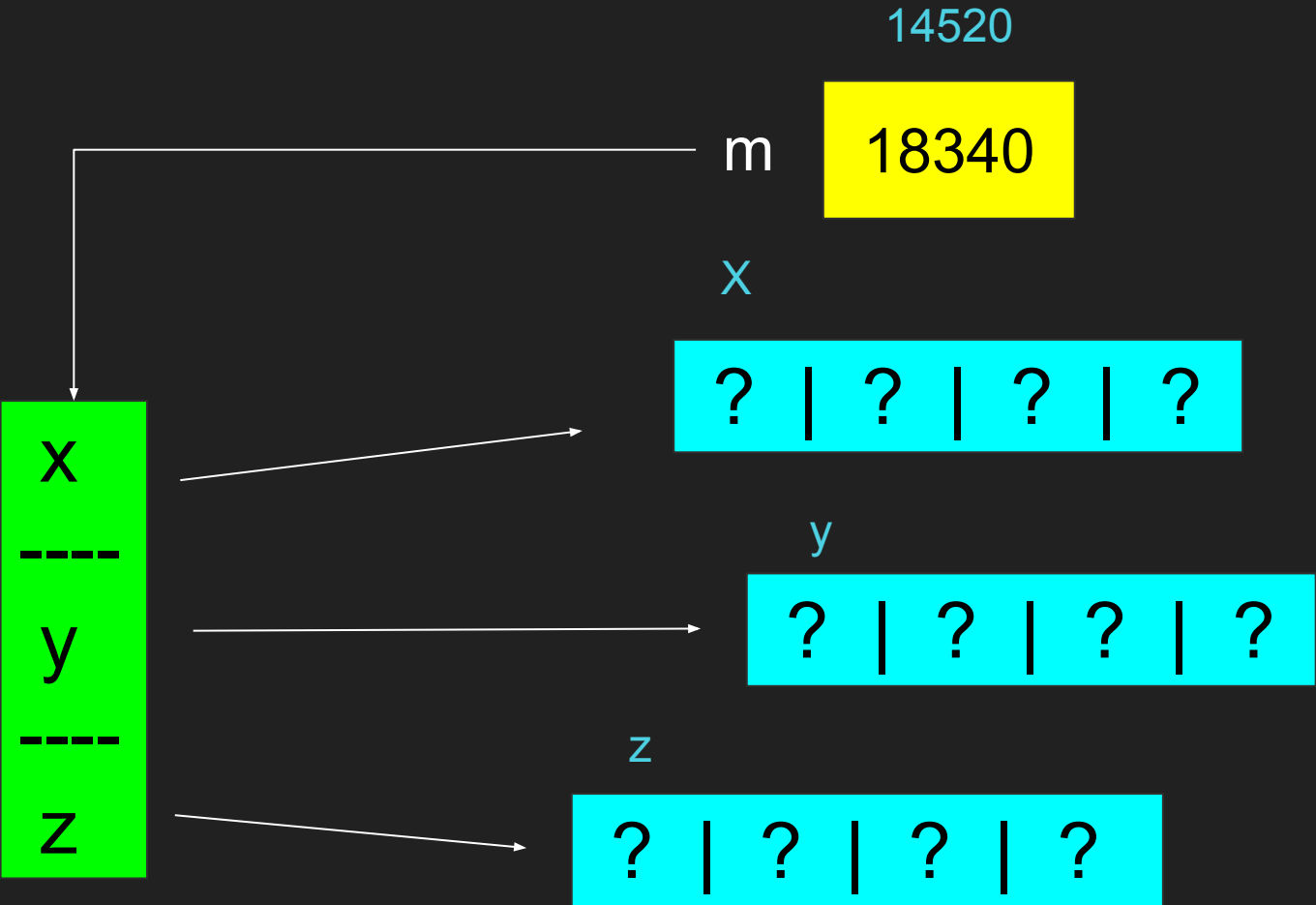
$m[i][j] == a[i * 4 + j]$

Matrizes (alocação dinâmica)

```
int main(){  
  
    int lin;  
  
    int ** m; // ponteiro de ponteiro? vetor de vetor?  
  
    m = (int **) malloc(3 * sizeof(int *));  
  
    for(lin = 0; lin < 3; lin++){ // lin: 0, 1, 2  
        m[lin] = (int *) malloc(4 * sizeof(int));  
    }  
  
    ...  
}
```

Matrizes

```
int ** m;
```



Matrizes (alocação dinâmica)

```
int main(){  
  
    int lin;  
  
    int ** m; // ponteiro de ponteiro? vetor de vetor?  
  
    m = (int **) malloc(3 * sizeof(int *));  
  
    int * a;  
  
    for(lin = 0; lin < 3; lin++){ // lin: 0, 1, 2  
        m[lin] = (int *) malloc(4 * sizeof(int));  
  
        if(lin == 0) a = m[lin];  
    }  
}
```

Matrizes

```
int ** m;
```

```
...
```

```
m[0][3];
```

```
a[3];
```

