Estructuras de Datos IV

Arreglos y Matrices

Jonatan Gómez Perdomo, Ph. D.
jgomezpe@unal.edu.co

Arles Rodríguez, Ph.D.
aerodriguezp@unal.edu.co

Camilo Cubides, Ph.D. (c)
eccubidesg@unal.edu.co

Carlos Andrés Sierra, M.Sc.
casierrav@unal.edu.co

Research Group on Artificial Life – Grupo de investigación en vida artificial – (Alife)
Computer and System Department
Engineering School
Universidad Nacional de Colombia

Agenda

Agenda

1. Arreglos (Vectores)
   - Conceptos
   - Lectura de un Arreglo
   - Ejemplos
2. Matrices
   - Conceptos
   - Ejemplos
3. Ejercicios y problemas

Definición

Un **arreglo** o **vector** es una tupla o lista de *n* elementos del mismo tipo (En este curso usaremos las listas para representar arreglos). A los elementos de un arreglo se les llama **componentes** del arreglo.

Ejemplo

- `[]`: Arreglo vacio.
- `[1, 0, 7, -2, 8]`: Un arreglo de enteros de 5 componentes.
- `['Radio', 'Lado']`: Un arreglo de cadenas de 2 componentes.
- `[1.3, 2.4, -3.0, 4.5]`: Un arreglo de reales de 4 componentes.
- `[True, False, False]`: Un arreglo de booleanos de 3 componentes.

Notación I

En general, un arreglo v se puede representar de la siguiente forma

$$v = [v_0, v_1, v_2, \ldots, v_{n-1}]$$

donde el arreglo está constituido por n componentes de un conjunto genérico $\mathcal{V}$. Si $v \in \mathcal{V}^n$, entonces el arreglo se dice que es n-dimensional o de tamaño n.

Ejemplo

El siguiente arreglo de tamaño 6 pertenece a $\mathbb{Z}^6$ y tiene una notación particular la cual es $0^6 = [0, 0, 0, 0, 0, 0]$

Notación II

En un vector, para referirse a una componente en particular, a ésta se le dice que es la *componente* en la posición i o la i-ésima componente, esto significa que el objeto es la componente ubicada en la posición i, se denota por la expresión v_i y se puede ubicar dentro del vector como se muestra a continuación

$$\begin{array}{c} \text{componente } i\text{-ésima} \\ \downarrow \\ [v_0, \ldots, v_i, \ldots, v_{n-1}] \end{array}$$

Notación III

Ejemplo

Para el vector

$$v = \left[-1,\ \tfrac{3}{4},\ -0.25,\ -\tfrac{1}{5},\ \sqrt{2},\ 0.0,\ \pi,\ \sqrt[3]{5},\ 0.\overline{9} \right]$$

de $\mathbb{R}^9$ se tiene que sus componentes son:

- $v_0 = -1$.
- $v_1 = \frac{3}{4}$.
- $v_2 = -0.25$.
- $v_3 = -\frac{1}{5}$.
- $v_4 = \sqrt{2}$.
- $v_5 = 0.0$.
- $v_6 = \pi$.
- $v_7 = \sqrt[3]{5}$.
- $v_8 = 0.\overline{9}$.

El conjunto de los arreglos

A partir de la notación de producto generalizado de un conjunto $\mathcal{V}$

$$\mathcal{V}^n = \underbrace{\mathcal{V} \times \mathcal{V} \times \mathcal{V} \times \cdots \times \mathcal{V}}_{n\text{-veces}}$$

se puede obtener el conjunto de los arreglos (vectores) $\mathcal{V}^*$, el cual se define como la unión de todos los productos cartesianos del conjunto $\mathcal{V}$, de la siguiente manera

$$\mathcal{V}^* = \bigcup_{n \in \mathbb{N}} \mathcal{V}^n$$

Agenda

1. Arreglos (Vectores)
   - Conceptos
   - Lectura de un Arreglo
   - Ejemplos
2. Matrices
   - Conceptos
   - Ejemplos
3. Ejercicios y problemas

Lectura de un arreglo

Se pueden combinar las funciones de leer un arreglo con la instrucción `for` para leer un arreglo de elementos separados por un carácter específico:

```
def lee_arreglo_enteros():
  return [int(x) for x in input("Arreglo:").split()]
print(lee_arreglo_enteros())
```

Lee un arreglo de enteros (usando `int`), separados por espacios.

```
def lee_arreglo_reales():
  return [float(x) for x in input("Arreglo:").split(",")]
print(lee_arreglo_reales())
```

Lee un arreglo de reales (usando `float`) separados por comas.

Agenda

1. Arreglos (Vectores)
   - Conceptos
   - Lectura de un Arreglo
   - Ejemplos
2. Matrices
   - Conceptos
   - Ejemplos
3. Ejercicios y problemas

Suma de las componentes de un arreglo

Si se tiene un arreglo de números, se puede definir la función `suma_arreglo` para calcular la suma de todos ellos:

Ejemplo

```
def suma_arreglo(A):
  s = 0
  for x in A:
    s += x
  return s

print(suma_arreglo([1,-3,4,11,6]))
```

Posición del máximo de un arreglo

Si se tiene un arreglo de números, se puede definir la función `pos_maximo` para calcular la posición donde se encuentra el máximo de ellos (el arreglo debe tener al menos una componente):

Ejemplo

```
def pos_maximo(A):
  m = 0
  for i in range(1,len(A)):
    if A[i] > A[m]:
      m = i
  return m

print(pos_maximo([1,-3,4,11,6]))
```

Agenda

1. Arreglos (Vectores)
   - Conceptos
   - Lectura de un Arreglo
   - Ejemplos
2. Matrices
   - Conceptos
   - Ejemplos
3. Ejercicios y problemas

Agenda

Definición

Una **matriz** es un arreglo rectangular de elementos del mismo tipo. En este curso usaremos arreglos de arreglos para representar matrices.

Ejemplo

La siguiente estructura rectangular de números enteros denotada por la expresión X es una matriz

$$X = \begin{bmatrix} 1 & 3 & 7 & -2 & 8 \\ 9 & 11 & 5 & 6 & 4 \\ 6 & -2 & -1 & 1 & 1 \end{bmatrix}$$

Notación I

En general, una matriz X se puede representar de la siguiente manera

$$X = \begin{bmatrix} x_{0,0} & x_{0,1} & x_{0,2} & \cdots & x_{0,(m-1)} \\ x_{1,0} & x_{1,1} & x_{1,2} & \cdots & x_{1,(m-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{(n-1),0} & x_{(n-1),1} & x_{(n-1),2} & \cdots & x_{(n-1),(m-1)} \end{bmatrix}$$

donde la matriz está compuesta por n **filas** y m **columnas**, a esta matriz se le dice que es de **tamaño** $n \times m$.

Notación II

A los objetos de la matriz se les llaman **componentes** o **entradas** de la matriz, y para referirse a una componente en particular, a ésta se le dice que es la **componente** en la posición (i, j), esto significa que la componente ubicada en la fila i y en la columna j, se denota por la expresión $X_{i,j}$ y se puede ubicar dentro de la matriz como se muestra a continuación

$$\begin{array}{cc}
 & \begin{array}{ccccc} \phantom{x_{(n-1),0}} & & \text{columna } j & & \phantom{x_{(n-1),(m-1)}} \\ & & \downarrow & & \end{array} \\
\begin{array}{c} \\ \\ \text{fila } i \longrightarrow \\ \\ \\ \end{array} &
\begin{bmatrix}
x_{0,0} & \cdots & x_{0,j} & \cdots & x_{0,(m-1)} \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
x_{i,0} & \cdots & \boxed{x_{i,j}} & \cdots & x_{i,(m-1)} \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
x_{(n-1),0} & \cdots & x_{(n-1),j} & \cdots & x_{(n-1),(m-1)}
\end{bmatrix}
\end{array}$$

Notación III

Para la matriz

$$X = \begin{bmatrix} 2 & -1 & \frac{3}{4} & -0.1 & e \\ -\frac{1}{5} & \sqrt{2} & 4 & 0.0 & 6 \\ 3.14 & \pi & \frac{1}{2} & \sqrt{3} & 3 \\ \sqrt[3]{5} & -10 & 0 & -5 & 0.\overline{9} \end{bmatrix}$$

de tamaño 4×5 se tiene que sus componentes son:

- $X_{0,0} = 2$
- $X_{0,1} = -1$
- $X_{0,2} = \frac{3}{4}$
- $X_{0,3} = -0.1$
- $X_{0,4} = e$
- $X_{1,0} = -\frac{1}{5}$
- $X_{1,1} = \sqrt{2}$
- $X_{1,2} = 4$
- $X_{1,3} = 0.0$
- $X_{1,4} = 6$
- $X_{2,0} = 3.14$
- $X_{2,1} = \pi$
- $X_{2,2} = \frac{1}{2}$
- $X_{2,3} = \sqrt{3}$
- $X_{2,4} = 3$
- $X_{3,0} = \sqrt[3]{5}$
- $X_{3,1} = -10$
- $X_{3,2} = 0$
- $X_{3,3} = -5$
- $X_{3,4} = 0.\overline{9}$

Notación IV

En Python, una matriz de tamaño $n \times m$ es un arreglo de n arreglos (llamados **filas**) de m elementos cada uno, teniendo que todos los arreglos son del mismo tipo:

$$n\left\{\underbrace{\begin{bmatrix} x_{0,0} & x_{0,1} & \cdots & x_{0,m-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{i,0} & x_{i,1} & \cdots & x_{i,m-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n-1,0} & x_{n-1,1} & \cdots & x_{n-1,m-1} \end{bmatrix}}_{m}\right. = \underbrace{\begin{bmatrix} \begin{bmatrix} x_{0,0} & x_{0,1} & \cdots & x_{0,m-1} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} x_{i,0} & x_{i,1} & \cdots & x_{i,m-1} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} x_{n-1,0} & x_{n-1,1} & \cdots & x_{n-1,m-1} \end{bmatrix} \end{bmatrix}}_{1}$$

Notación V

A partir del concepto de arreglo, se puede ahora definir el *conjunto de las matrices* $\mathbb{T}^{**}$ como la unión de todos los productos cartesianos del conjunto de los arreglos del conjunto $\mathbb{T}$, de la siguiente manera

$$\mathbb{T}^{**} = \bigcup_{n \in \mathbb{N}} \left(\bigcup_{m \in \mathbb{N}} \mathbb{T}^m \right)^n$$

Un elemento genérico del conjunto $\mathbb{T}^{**}$ es de la forma $(\mathbb{T}^m)^n$, donde n es el número de filas y m es el número de columnas. Para abreviar, de aquí en adelante se utilizará la notación

$$(\mathbb{T}^m)^n \Leftrightarrow \mathbb{T}^{n \times m}.$$

Agenda

Cuadrados de elementos de una matriz I

Una función general que permite construir una nueva matriz que contiene el cuadrado de cada componente de una matriz dada es

Ejemplo

$$
\begin{aligned}
\mathit{cuadrados\_matriz} : \mathbb{Z}^{n\times m} &\to \mathbb{Z}^{n\times m} \\
(X) &\mapsto Y, \quad \text{donde} \quad Y_{i,j} = X_{i,j}^2, \\
&\qquad\qquad\qquad\qquad \forall i = 0, 1, \ldots, n-1, \\
&\qquad\qquad\qquad\qquad \forall j = 0, 1, \ldots, m-1
\end{aligned}
$$

Cuadrados de elementos de una matriz II

Una función general que permite construir una nueva matriz que contiene el cuadrado de cada componente de una matriz dada es

Ejemplo

```
def cuadrados_matriz(X):
  Y = []
  for i in range(len(X)):
    fila = []
    for j in range(len(X[i])):
      fila.append(X[i][j]*X[i][j])
    Y.append(fila)
  return Y
print(cuadrados_matriz([[1,-3, 2],[4,11,-1]]))
```

Diagonal de una matriz cuadrada I

Una función general que permite obtener un arreglo con los elementos que están en la diagonal de una matriz cuadrada (mismo número de filas y columnas) es:

Ejemplo

$$
\begin{aligned}
diagonal\_matriz : \mathbb{Z}^{n \times n} &\to \mathbb{Z}^{n} \\
(X) &\mapsto Y, \quad \text{donde} \quad Y_i = X_{i,i}, \\
&\qquad\qquad\qquad\quad \forall i = 0, 1, \ldots, n-1
\end{aligned}
$$

Diagonal de una matriz cuadrada II

Una función general que permite obtener un arreglo con los elementos que están en la diagonal de una matriz cuadrada (mismo número de filas y columnas) es:

Ejemplo

```
def diagonal_matriz(X):
  Y = []
  for i in range(len(X)):
    Y.append(X[i][i])
  return Y
print(diagonal_matriz([[1,-3],[4,11]]))
```

Simetría de una matriz I

Una función general que permite determinar si una matriz cuadrada es simétrica o no, es:

Ejemplo

$$matriz\_simetrica : \mathbb{Z}^{n\times n} \to \mathbb{B}$$

$$(X) \mapsto \bigwedge_{\substack{i=0\\ j=i+1}}^{\substack{n-1\\ n-2}} X_{i,j} \equiv X_{j,i}$$

Simetría de una matriz II

Una función general que permite determinar si una matriz cuadrada es simétrica o no, es:

Ejemplo

```
def matriz_simetrica(X):
  bandera = True
  for i in range(len(X)-1):
    for j in range(i+1,len(X)):
      bandera = bandera and (X[i][j] == X[j][i])
  return bandera
print(matriz_simetrica([[1,-3],[4,11]]))
print(matriz_simetrica([["A","B"],["B","A"]]))
```

Agenda

Problemas de arreglos I

Problemas

1. Desarrollar un algoritmo que calcule el promedio de un arreglo de reales.
2. Desarrollar un algoritmo que calcule el producto punto de dos arreglos de números enteros (reales) de igual tamaño. Sean $v = [v_0, v_1, \ldots, v_{n-1}]$ y $w = [w_0, w_1, \ldots, w_{n-1}]$ dos arreglos, el producto de v y w (notado $v \cdot w$) es el número: $v_0 * w_0 + v_1 * w_1 + \cdots + v_{n-1} * w_{n-1}$.
3. Desarrollar un algoritmo que calcule el producto directo de dos arreglos de números reales de igual tamaño. Sean $v = [v_0, v_1, \ldots, v_{n-1}]$ y $w = [w_0, w_1, \ldots, w_{n-1}]$ dos arreglos, el producto directo de v y w (notado $v * w$) es el vector: $[v_0 * w_0, v_1 * w_1, \ldots, v_{n-1} * w_{n-1}]$.

Problemas de arreglos II

Problemas

4. Desarrollar un algoritmo que determine la mediana de un arreglo de enteros. La mediana es el número que queda en la mitad del arreglo después de ser ordenado.
5. Hacer un algoritmo que deje al final de un arreglo de números todos los ceros que aparezcan en dicho arreglo.

Ejemplo

vector original: $[1, 6, 0, 7, -3, 8, 0, -2, 11]$
vector salida: $[1, 6, 7, -3, 8, -2, 11, 0, 0]$

Ejemplo

vector original: $[0, 11, 36, 10, 0, 17, -23, 81, 0, 0, 12, 11, 0]$
vector salida: $[11, 36, 10, 17, -23, 81, 12, 11, 0, 0, 0, 0, 0]$

Problemas de matrices I

Problemas

1. Desarrollar un algoritmo que permita sumar dos matrices de números reales (enteros).
2. Desarrollar un algoritmo que permita multiplicar dos matrices de números reales (enteros).
3. Desarrollar un programa que sume los elementos de una columna dada de una matriz.
4. Desarrollar un programa que sume los elementos de una fila dada de una matriz.

Problemas de matrices II

Problemas

5. Desarrollar un algoritmo que determine si una matriz es mágica. Se dice que una matriz cuadrada es mágica si la suma de cada una de sus filas, de cada una de sus columnas y de cada diagonal es igual.

Ejemplo

8	1	6
3	5	7
4	9	2

Ejemplo

1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1