

MATRICES

109

Matriz \equiv [fila ; fila ; ... fila]

fila \equiv numero , ... , numero

NOTA

Hay que comprobar
dimensiones al leer la
matriz !!

No queremos solo leer matrices.

Queremos sentencias con ellas. Ej:

```
a = [ ..... ] ;
```

```
print(a) ;
```

```
b = traspuesta(a) ;
```

```
print( a+b ) ;
```

Queremos :

- i) Declaración de matrices $a = [\dots] ;$
- ii) Impresión matrices $print(a) ;$
- iii) Operaciones
- iv) Comprobación de Dimensiones en lectura
- v) Comprobación de Dimensiones en funciones y operaciones
- vi) Definición y uso de variables
- vii) Compatibilidad de formatos ($A * [\dots]$)
- viii) Lectura Jass: $1\ 1\ 1.2\ 3\ 4\ 1\ 1.2\ 3\ 4$

Vamos a hacerlo todo poco a poco. Probando y luego pasando al siguiente requisito.

i) Queremos declaración e impresión de
y matrices. Antes de ello, queremos un
programa que lea varias líneas

Con esto
aseguramos
leer varias líneas

axioma → axioma línea
| línea

Cómo es una línea?

Primera
sentencia a probar

De momento será

línea → print (matriz) P4C

¿ Y una matriz ?

Matriz → [listaFilas]

listaFilas → listaFilas P4C fila

listaFilas → fila

fila → fila como número

fila → número

ArrayList<ArrayList<Integer>>

ArrayList<Integer>

Hasta aquí la Matrices V1. cup

Ahora queremos también los jugadores en
Jueces por ejemplo,
en este caso

trasputa
inurse
ad junta

Pero eso no son verí que otras matrices.

luego

Matriz \rightarrow transpuesta AP matriz CP

Matriz \rightarrow adjunta AP matriz CP

Matriz \rightarrow inversa AP matriz CP.

Ojo!! Que hay que comprobar antes que se
pueden calcular y manejar errores!!

\downarrow \rightarrow Es va en las acciones, (reglas semánticas)

para inversa y adjunta debe ser cuadrada.

iii)

También se requieren operaciones
con matrices. Qué operaciones queremos?

1) Mult matricial $m \times n$ $p \times q$ en $n=p$

2) Suma Matricial $m \times n$ $m \times n$

Comprobaciones

matriz \rightarrow matriz por matriz

matriz \rightarrow matriz mas matriz

3) Suma Escalar Matriz \rightarrow matriz mas numero

4) Prod Escalar igual como por

5) Matriz \rightarrow (Matriz)

producto Escalar (Positivo) \leftarrow Yo solo he
Suma Escalar (Positivo y Neg) \leftarrow implementado

Ojo: los numeros
pueden
ser
positivos, negativos
o double

iv-v) Se han hecho antes las comprobaciones

vi) Definición y uso de variables.

Ahora una línea puede ser una asignación
línea \rightarrow IDENT = matA ;



Además debe guardar la
matriz en la tabla de símbolos



guardamos realmente
en memoria la
matriz.

Pero qo, eso era para asignar, ahora
también puede hacerse que una matriz sea
un ident, es decir, si escrito $A = B$
y A y B se han asignado antes, debe
hacerse, ie,

matriz \rightarrow IDENT:5 { result = ts.get(5) }

Por último: notación javier

{ ... , ... }

matriz \rightarrow ALL listaFilaJavier CLL

listaFilaJavier \rightarrow listaFilaJavier como ALL fila CLL
| ALL fila CLL