

Práctica 2: Calculadora de Matrices

Archundia Bazán Aarón Antonio, Guerrero Velez Eliseo Milton, Hernández Vázquez Cesar Arturo
Unidad Profesional Interdisciplinaria en Ingeniería y Tecnologías Avanzadas del I.P.N.
Ingeniería Biónica, Programación Orientada a Objetos
(Dated: 23 de octubre de 2020)

I. PLANTEAMIENTO DEL PROBLEMA

El problema consiste en desarrollar una calculadora de matrices que trabaje con las siguientes operaciones para matrices bidimensionales de cualquier tamaño:

- Suma
- Resta
- Multiplicación
- Transpuesta

Donde el tamaño de la matriz sea dado por el usuario y respete las propiedades matrices así como las reglas para cada uno de los métodos antes propuestos.

II. PROPUESTA DE SOLUCIÓN

Para dar una solución a este problema hay que revisar cuales con las que se tiene que trabajar este sistema. En este caso las propiedades para las operaciones con matrices.

Matriz:

Una matriz A de $m \times n$ un arreglo rectangular mn números dispuestos en m renglones y n columnas.

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

Suma de matrices:

La suma de dos matrices se define únicamente cuando son del mismo tamaño. La suma de A y B está dada por A+B:

$$A + B = \begin{pmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \cdots & a_{1n} + b_{1n} \\ a_{21} + b_{21} & a_{22} + b_{22} & \cdots & a_{2n} + b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} + b_{m1} & a_{m2} + b_{m2} & \cdots & a_{mn} + b_{mn} \end{pmatrix}$$

Resta de matrices:

La resta de dos matrices se define únicamente cuando son del mismo tamaño. La resta de A y B está dada por A-B:

$$A - B = \begin{pmatrix} a_{11} - b_{11} & a_{12} - b_{12} & \cdots & a_{1n} - b_{1n} \\ a_{21} - b_{21} & a_{22} - b_{22} & \cdots & a_{2n} - b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} - b_{m1} & a_{m2} - b_{m2} & \cdots & a_{mn} - b_{mn} \end{pmatrix}$$

Multiplicación de Matrices: Sean dos matrices AB las propiedades que deben cumplir son las siguientes:

$$C = AB = (c_{ij})m \times n$$

Donde cada elemento c_{ij} está definido por:

$$c_{ij} = \sum_{r=1}^n a_{ir}b_{rj}$$

$$C = A * B = \begin{pmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \cdots & a_{1n} + b_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} + b_{m1} & a_{m2} + b_{m2} & \cdots & a_{mn} + b_{mn} \end{pmatrix} * \begin{pmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \cdots & a_{1p} + b_{1p} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} + b_{n1} & a_{n2} + b_{n2} & \cdots & a_{np} + b_{np} \end{pmatrix}$$

$$C = \begin{pmatrix} a_{11}b_{11} + \cdots + a_{1n}b_{n1} & \cdots & a_{11}b_{np} + \cdots + a_{1n}b_{np} \\ \vdots & \ddots & \vdots \\ a_{m1}b_{11} + \cdots + a_{mn}b_{n1} & \cdots & a_{m1}b_{np} + \cdots + a_{mn}b_{np} \end{pmatrix}$$

Dos matrices se pueden multiplicar únicamente si el número de columnas de la primera matriz es igual a la segunda. De otro modo, los vectores que forman el renglón i en A y la columna j de B no tendrán el mismo número de componentes y el producto punto en la ecuación Dicho de otro modo, las matrices A y B serán incompatibles bajo la multiplicación.

Transpuesta de una matriz: Sea $A = (a_{ij})$ una matriz de $m \times n$. Entonces la transpuesta de A , que se escribe A^T , es la matriz de $n \times m$ que se obtiene al intercambiar los renglones por las columnas de A . De manera breve, se puede escribir $A^T (a_{ij})$. En otras palabras

$$\text{Si } A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}, \text{ entonces } A^T = \begin{pmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & \cdots & a_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{mn} \end{pmatrix}$$

Simplemente se coloca el renglón i de A como la columna i de A^T y la columna j de A como el renglón j de A^T .

Por lo tanto con los datos anteriores la propuesta de solución fue hacer una función main con pocas líneas de código que contenga un switch que pregunte la operación a realizar, una vez seleccionada ejecutará la operación llamando a la función creada para cada caso de cálculo de matrices, al finalizar un do-while determinará por medio de una entrada si el programa continua corriendo o si se quiere salir.

III. ANÁLISIS Y DISEÑO

A. Diagrama de flujo función main()

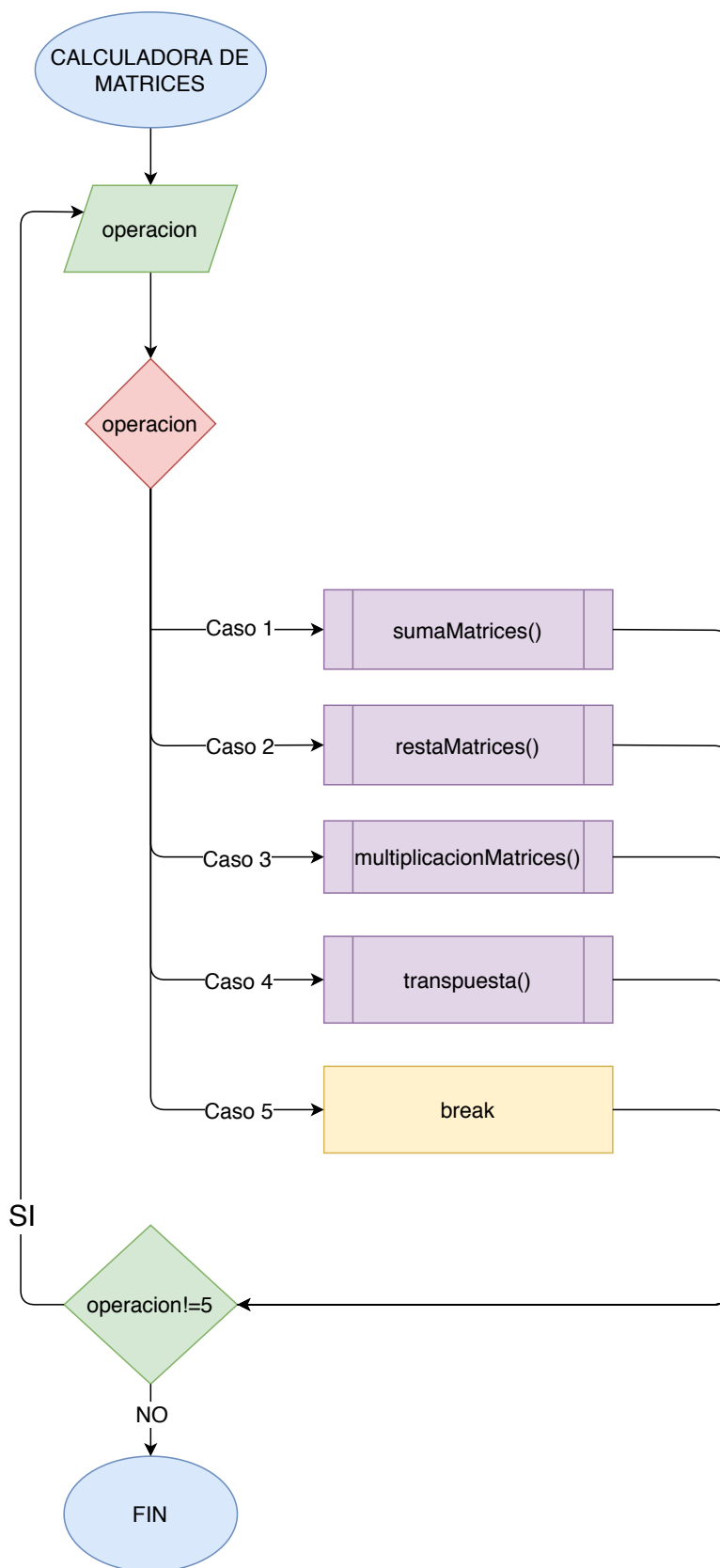


Diagrama 1: Función Main()

B. Diagrama función sumaMatrices ()

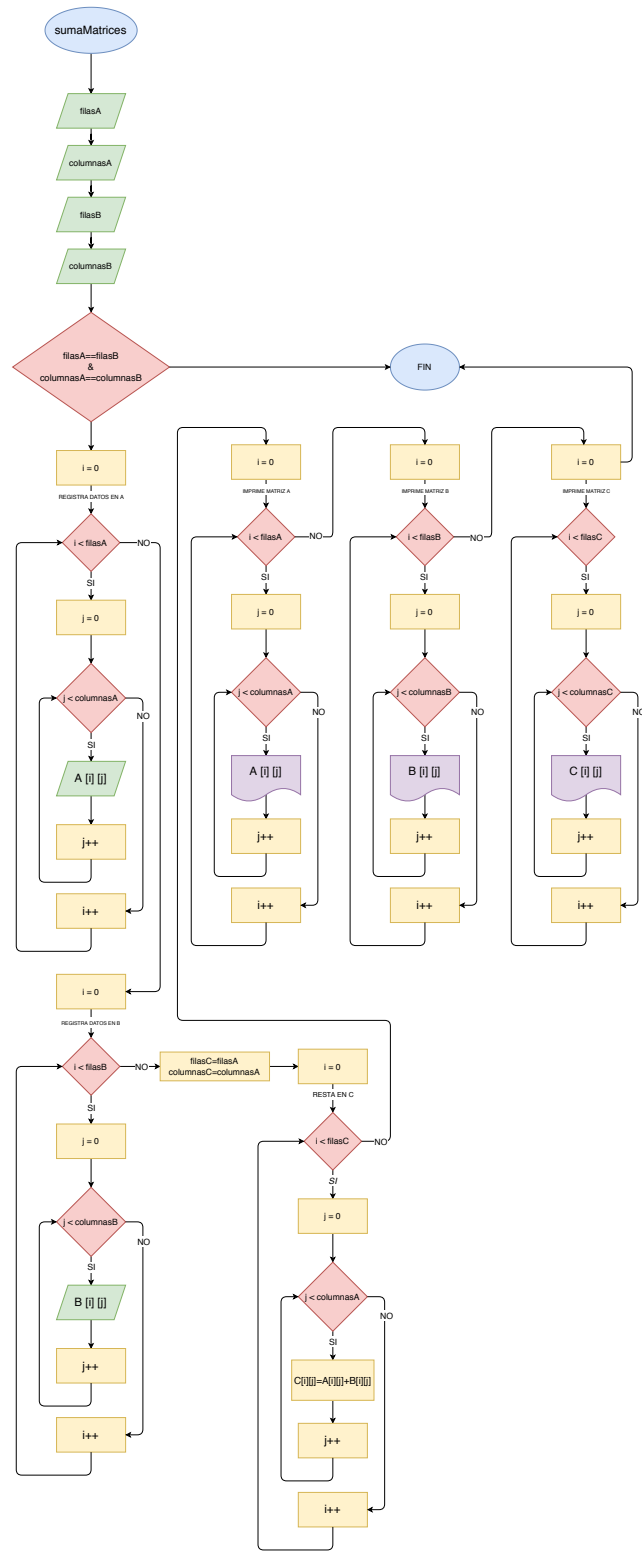


Diagrama 2: Función sumaMatrices()

C. Diagrama función restaMatrices ()

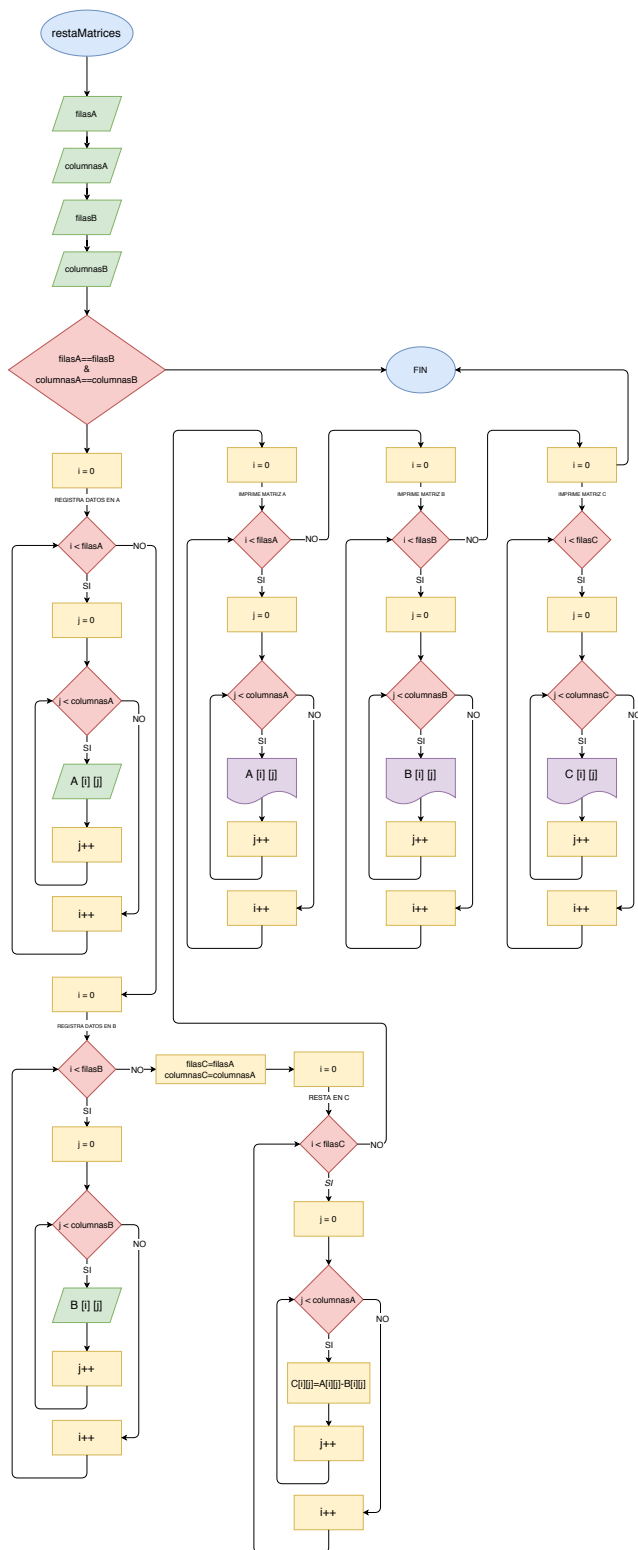


Diagrama 3: Función restaMatrices()

D. Diagrama función multiplicacionMatrices ()

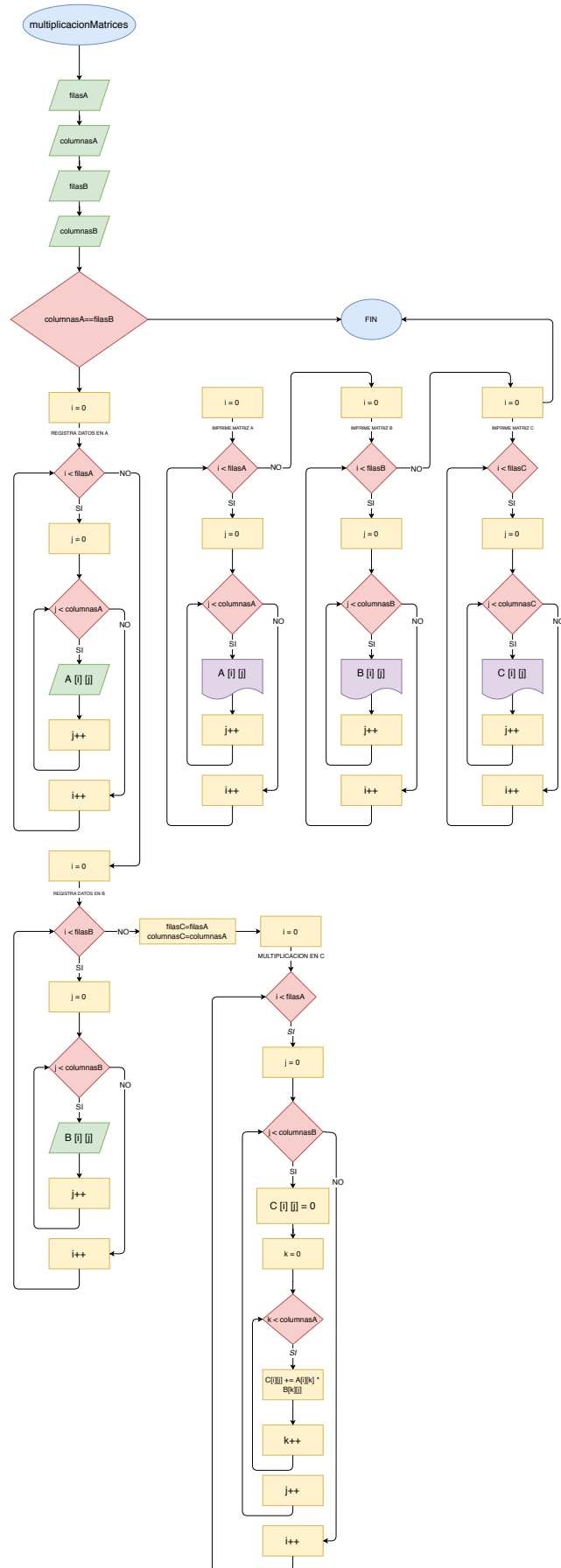
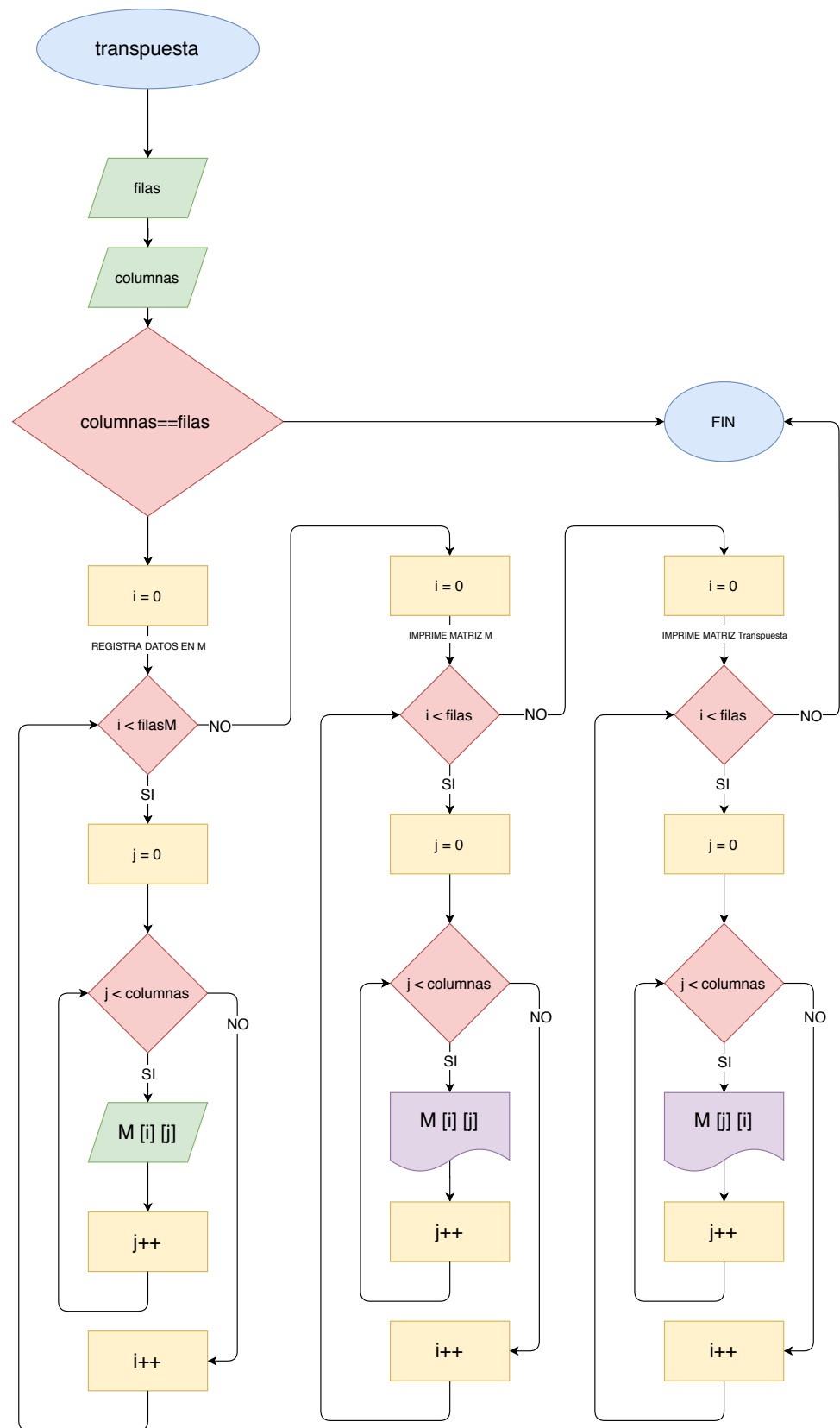


Diagrama 4: Función multiplicacionMatrices()

E. Diagrama función transpuesta ()

Diagrama 5: Función `transpuesta()`

IV. IMPLEMENTACIÓN Y PRUEBAS

V. CÓDIGO FUENTE COMENTADO

```

1 //Pr ctica 2: CALCULADORA DE MATRICES
2 //ARCHUNDIA BAZ N AAR N ANTONIO, GUERRERO V LEZ ELISEO MILTON, HERN NDEZ V ZQUEZ C SAR ATURO
3 #include <iostream>
4 #include <locale>
5
6 using namespace std;
7
8 void operacionesDisponibles();
9 void sumaMatrices();
10 void restaMatrices();
11 void multiplicacionMatrices();
12 void transpuesta();
13
14 int main()
15 {
16     setlocale(LC_ALL, "");
17     int operacion{0};
18     do
19     {
20         operacionesDisponibles();
21         cout<<"Seleccione la operaci n que desea realizar: ";
22         cin>>operacion;
23         cout<<endl;
24
25         switch(operacion)
26         {
27             case 1:
28                 sumaMatrices();
29                 break;
30             case 2:
31                 restaMatrices();
32                 break;
33             case 3:
34                 multiplicacionMatrices();
35                 break;
36             case 4:
37                 transpuesta();
38                 break;
39             case 5:
40                 break;
41         }
42     } while(operacion!=5);
43     return 0;
44 }
45
46 void operacionesDisponibles()
47 {
48     cout<<"CALCULADORA DE MATRICES"<<endl;
49     cout<<"BIENVENIDO..."<<endl;
50     cout<<" "<<endl;
51     cout<<"Listado de Operaciones de Matrices Disponibles:"<<endl;
52     cout<<"-----"<<endl;
53     cout<<" "<<endl;
54     cout<<"1. SUMA (A+B) "<<endl;
55     cout<<"2. RESTA (A-B)"<<endl;
56     cout<<"3. MULTIPLICACI N (A*B)"<<endl;
57     cout<<"4. MATRIZ TRANSPUESTA "<<endl;
58     cout<<"5. SALIR DEL PROGRAMA "<<endl;
59     cout<<" "<<endl;
60 }
61
62 void sumaMatrices() //Funci n para la suma de matrices A y B
63 {
64     //Declaraci n de variables para las filas y columnas de las matrices A y B
65     int filasA{0}, columnasA{0};

```



```

66  int filasB{0}, columnasB{0};
67
68  //Se imprime encabezado para la suma
69  cout<<" "<<endl;
70  cout<<"-----SUMA DE MATRICES-----"<<endl;
71  cout<<" "<<endl;
72  cout<<"|||  NOTA !: Las Matrices deben ser del mismo tama o |||"<<endl;
73  cout<<"-----"<<endl;
74
75
76  //Se le solicita al usuario indicar filas y columnas de la matriz A
77  cout<<" "<<endl;
78  cout<<"* Inserte n mero de filas para la matriz A: ";
79  cin>>filasA;
80  cout<<endl;
81  cout<<"* Inserte n mero de columnas para la matriz A: ";
82  cin>>columnasA;
83  cout<<endl;
84
85  //Se le solicita al usuario indicar filas y columnas de la matriz B
86  cout<<"* Inserte n mero de filas para la matriz B: ";
87  cin>>filasB;
88  cout<<endl;
89  cout<<"* Inserte n mero de columnas para la matriz B: ";
90  cin>>columnasB;
91  cout<<endl;
92
93  if(filasA==filasB&columnasA==columnasB) //Si las matrices A y B son del mismo tama o entonces
94  {
95      //Inserta los valores para cada elemento de la matriz A
96      cout<<" "<<endl;
97      cout <<"DATOS PARA LA MATRIZ A: "<<endl;
98      int A[filasA][columnasA];
99      for(size_t i{0}; i<filasA; i++)
100      {
101          for(size_t j{0}; j<columnasA; j++)
102          {
103              cout <<"Matriz["<<i<<"]["<<j<<": ";
104              cin >> A[i][j];
105          }
106      }
107
108      //Inserta los valores para cada elemento de la matriz B
109      cout<<" "<<endl;
110      cout <<"DATOS PARA LA MATRIZ B: "<<endl;
111      int B[filasB][columnasB];
112
113      for(size_t i{0}; i<filasB; i++)
114      {
115          for(size_t j{0}; j<columnasB; j++)
116          {
117              cout <<"Matriz["<<i<<"]["<<j<<": ";
118              cin >> B[i][j];
119          }
120      }
121
122      //Suma en la Matriz C
123      int filasC=filasA;
124      int columnasC=columnasA;
125      int C[filasC][columnasC];
126      for(size_t i{0};i<filasC;i++)
127      {
128          for(size_t j{0}; j<columnasC;j++)
129          {
130              C[i][j] = A[i][j] + B[i][j];
131          }
132      }
133
134

```

```

135 //Imprime Matriz A
136 cout<<"-----"<<endl;
137 cout<<" "<<endl;
138 cout<<"MATRIZ A"<<endl;
139 cout<<" "<<endl;
140 for(size_t i{0};i<filasA;i++)
141 {
142     cout <<" | ";
143     for(size_t j{0}; j<columnasA; j++)
144     {
145         cout<<A[i][j]<<" ";
146     }
147     cout <<"|"<<endl;
148 }
149 cout<<" "<<endl;
150
151
152 //Imprime Matriz B
153 cout<<"MATRIZ B"<<endl;
154 cout<<" "<<endl;
155 for(size_t i{0};i<filasB;i++)
156 {
157     cout <<" | ";
158     for(size_t j{0}; j<columnasB; j++)
159     {
160         cout<<B[i][j]<<" ";
161     }
162     cout <<"|"<<endl;
163 }
164 cout<<" "<<endl;
165
166
167 //Imprime Matriz C
168 cout<<"MATRIZ C = A+B"<<endl;
169 cout<<" "<<endl;
170
171 for(size_t i{0};i<filasC;i++)
172 {
173     cout <<" | ";
174     for(size_t j{0}; j<columnasC; j++)
175     {
176         cout<<C[i][j]<<" ";
177     }
178     cout <<"|"<<endl;
179 }
180 cout<<" "<<endl;
181
182
183 }
184 else //Imprime un aviso que no se pudo llevar a cabo la operaci n por la diferencia de
185 dimensiones
186 {
187     cout<<" "<<endl;
188     cout<<" AVISO !!!!!"<<endl;
189     cout<<"-----"<<endl;
190     cout<<" "<<endl;
191     cout<<" NO SE PUEDE REALIZAR LA OPERACI N"<<endl;
192     cout<<" "<<endl;
193     cout<<" NO TIENEN LA MISMA DIMENSI N"<<endl;
194     cout<<"-----"<<endl;
195     cout<<" "<<endl;
196 }
197 }
198
199
200
201 void restaMatrices() //Funci n para la resta de matrices A y B
202 {
203     //Declaraci n de variables para las filas y columnas de A y B

```

```

204 int filasA{0}, columnasA{0};
205 int filasB{0}, columnasB{0};
206
207 //Se imprime encabezado para la resta
208 cout<<" "<<endl;
209 cout<<"-----RESTA DE MATRICES-----"<<endl;
210 cout<<" "<<endl;
211 cout<<"|||  NOTA !: Las Matrices deben ser del mismo tama o |||"<<endl;
212 cout<<"-----"<<endl;
213
214
215 //Se le solicita al usuario ndicar filas y columnas de la matriz A
216 cout<<" "<<endl;
217 cout<<"* Inserte n mero de filas para la matriz A: ";
218 cin>>filasA;
219 cout<<endl;
220 cout<<"* Inserte n mero de columnas para la matriz A: ";
221 cin>>columnasA;
222 cout<<endl;
223
224 //Se le solicita al usuario ndicar filas y columnas de la matriz B
225 cout<<"* Inserte n mero de filas para la matriz B: ";
226 cin>>filasB;
227 cout<<endl;
228 cout<<"* Inserte n mero de columnas para la matriz B: ";
229 cin>>columnasB;
230 cout<<endl;
231
232
233 if(filasA==filasB&columnasA==columnasB) //Si la matriz A y B tienen el mismo tama o entonces se
    realiza la operaci n
234 {
235     //Inserta los valores para cada elemento de la matriz B
236     cout<<" "<<endl;
237     cout <<"DATOS PARA LA MATRIZ A: "<<endl;
238     int A[filasA][columnasA];
239     for(size_t i{0}; i<filasA; i++)
240     {
241         for(size_t j{0}; j<columnasA; j++)
242         {
243             cout <<"Matriz["<<i<<"]["<<j<<"]: ";
244             cin >> A[i][j];
245         }
246     }
247
248     //Inserta los valores para cada elemento de la matriz B
249     cout<<" "<<endl;
250     cout <<"DATOS PARA LA MATRIZ B: "<<endl;
251     int B[filasB][columnasB];
252
253     for(size_t i{0}; i<filasB; i++)
254     {
255         for(size_t j{0}; j<columnasB; j++)
256         {
257             cout <<"Matriz["<<i<<"]["<<j<<"]: ";
258             cin >> B[i][j];
259         }
260     }
261
262     //Suma en la Matriz C
263     int filasC=filasA;
264     int columnasC=columnasA;
265     int C[filasC][columnasC];
266     for(size_t i{0};i<filasC;i++)
267     {
268         for(size_t j{0}; j<columnasC;j++)
269         {
270             C[i][j] = A[i][j] - B[i][j];
271         }
272     }

```

```

273
274 //Imprime Matriz A
275 cout<<"-----" <<endl;
276 cout<<" " <<endl;
277 cout<<"MATRIZ A" <<endl;
278 cout<<" " <<endl;
279 for(size_t i{0}; i<filasA; i++)
280 {
281     cout <<" | ";
282     for(size_t j{0}; j<columnasA; j++)
283     {
284         cout<<A[i][j]<<" ";
285     }
286     cout <<"|" <<endl;
287 }
288 cout<<" " <<endl;
289
290
291 //Imprime Matriz B
292 cout<<"MATRIZ B" <<endl;
293 cout<<" " <<endl;
294 for(size_t i{0}; i<filasB; i++)
295 {
296     cout <<" | ";
297     for(size_t j{0}; j<columnasB; j++)
298     {
299         cout<<B[i][j]<<" ";
300     }
301     cout <<"|" <<endl;
302 }
303 cout<<" " <<endl;
304
305
306
307 //Imprime Matriz C
308 cout<<"MATRIZ C = A-B" <<endl;
309 cout<<" " <<endl;
310
311 for(size_t i{0}; i<filasC; i++)
312 {
313     cout <<" | ";
314     for(size_t j{0}; j<columnasC; j++)
315     {
316         cout<<C[i][j]<<" ";
317     }
318     cout <<"|" <<endl;
319 }
320 cout<<" " <<endl;
321
322
323 }
324 else //Imprime un aviso que no se pudo llevar a cabo la operaci n por la diferencia de
dimensiones
325 {
326     cout<<" " <<endl;
327     cout<<" AVISO !!!!!" <<endl;
328     cout<<"-----" <<endl;
329     cout<<" " <<endl;
330     cout<<" NO SE PUEDE REALIZAR LA OPERACI N" <<endl;
331     cout<<" " <<endl;
332     cout<<" NO TIENEN LA MISMA DIMENSI N" <<endl;
333     cout<<"-----" <<endl;
334     cout<<" " <<endl;
335 }
336 }
337
338
339 void multiplicacionMatrices() //Funci n para la multiplicaaci n de matrices A y B
340 {
341     //Declaracion de variables para las filas y columnas de la matriz A y B

```

```

342 int filasA{0}, columnasA{0};
343 int filasB{0}, columnasB{0};
344
345 //Se imprime encabezado para la multiplicaci n
346 cout<<" "<<endl;
347 cout<<"-----MULTIPLICACI N DE MATRICES-----"<<endl;
348 cout<<" "<<endl;
349 cout<<"|||  NOTA !: EL NUMERO DE COLUMNAS DE A  |||"<<endl;
350 cout <<"||| DEBE SER EL MISMO NUMERO DE FILAS DE B  |||"<<endl;
351 cout<<"-----" <<endl;
352
353
354 //Se le solicita al usuario Indicar filas y columnas de la matriz A
355 cout<<" "<<endl;
356 cout<<"* Inserte n mero de filas para la matriz A: ";
357 cin>>filasA;
358 cout<<endl;
359 cout<<"* Inserte n mero de columnas para la matriz A: ";
360 cin>>columnasA;
361 cout<<endl;
362
363 //Inserta los valores para cada elemento de la matriz A
364 cout<<" "<<endl;
365     cout <<"DATOS PARA LA MATRIZ A: "<<endl;
366     int A[filasA][columnasA];
367     for(size_t i{0}; i<filasA; i++)
368     {
369         for(size_t j{0}; j<columnasA; j++)
370         {
371             cout <<"Matriz["<<i<<"]["<<j<<"]: ";
372             cin >> A[i][j];
373         }
374     }
375
376
377 //Se le solicita al usuario indicar filas y columnas de la matriz B
378 cout<<"* Inserte n mero de filas para la matriz B: ";
379 cin>>filasB;
380 cout<<endl;
381 cout<<"* Inserte n mero de columnas para la matriz B: ";
382 cin>>columnasB;
383 cout<<endl;
384
385 //Inserta los valores para cada elemento de la matriz B
386 cout<<" "<<endl;
387     cout <<"DATOS PARA LA MATRIZ B: "<<endl;
388     int B[filasB][columnasB];
389
390     for(size_t i{0}; i<filasB; i++)
391     {
392         for(size_t j{0}; j<columnasB; j++)
393         {
394             cout <<"Matriz["<<i<<"]["<<j<<"]: ";
395             cin >> B[i][j];
396         }
397     }
398
399
400 //Imprime Matriz A
401 cout<<"-----" <<endl;
402 cout<<" "<<endl;
403 cout<<"MATRIZ A"<<endl;
404 cout<<" "<<endl;
405     for(size_t i{0};i<filasA;i++)
406     {
407         cout <<" | ";
408         for(size_t j{0}; j<columnasA; j++)
409         {
410             cout<<A[i][j]<<" ";
411         }

```

```

412     cout << "|" << endl;
413 }
414 cout << " " << endl;
415
416 //Imprime Matriz B
417 cout << "MATRIZ B" << endl;
418 cout << " " << endl;
419 for (size_t i{0}; i < filasB; i++)
420 {
421     cout << " | ";
422     for (size_t j{0}; j < columnasB; j++)
423     {
424         cout << B[i][j] << " ";
425     }
426     cout << "|" << endl;
427 }
428 cout << " " << endl;
429
430 //Declaraci n Matriz C
431 int filasC = filasA;
432 int columnasC = columnasA;
433 int C[filasC][columnasC];
434
435
436
437 //Multiplicaci n de las Matrices
438 if (columnasA == filasB) //Si las columnas de A y las filas de A entonces realiza la operaci n
439 {
440     for (int i{0}; i < filasA; i++)
441     {
442         for (int j{0}; j < columnasB; j++)
443         {
444             C[i][j] = 0;
445             for (int k = 0; k < columnasA; k++)
446             {
447                 C[i][j] += A[i][k] * B[k][j];
448             }
449         }
450     }
451 }
452
453 //Imprime Matriz C
454 cout << " " << endl;
455 cout << "C = A*B:";
456 cout << " " << endl;
457
458 for (size_t i{0}; i < filasC; i++)
459 {
460     cout << " | ";
461     for (size_t j{0}; j < columnasC; j++)
462     {
463         cout << C[i][j] << " ";
464     }
465     cout << "|" << endl;
466 }
467 cout << " " << endl;
468 }
469 else //Imprime un aviso que no se pudo llevar a cabo la operaci n por la diferencia de
470     dimensiones en columnas de B y filas de A
471 {
472     cout << " " << endl;
473     cout << "                                AVISO          !!!!!" << endl;
474     cout << "-----" << endl;
475     cout << " " << endl;
476     cout << "                                NO SE PUEDE REALIZAR LA OPERACI N" << endl;
477     cout << " " << endl;
478     cout << "                                EL N MERO DE COLUMNAS DE LA MATRIZ A" << endl;
479     cout << "                                NO ES EL MISMO N MERO DE FILAS DE LA MATRIZ B" << endl;
480     cout << "-----" << endl;

```

```

481     cout<<" "<<endl;
482 }
483 }
484
485 void transpuesta() //Funci n para sacar transpuesta de una Matriz
486 {
487     //Declara variables para guardar filas y columnas de la matriz
488     int filas{0}, columnas{0};
489
490     //Se imprime encabezado para la Transpuesta
491     cout<<" "<<endl;
492     cout<<"-----TRANSPUESTA DE MATRICES-----"<<endl;
493     cout<<" "<<endl;
494     cout<<"||| NOTA !: EL NUMERO DE COLUMNAS |||"<<endl;
495     cout <<"||| DEBE SER EL MISMO NUMERO DE FILAS |||"<<endl;
496     cout<<"-----"<<endl;
497
498
499     //Indicar filas y columnas de la matriz
500     cout<<" "<<endl;
501     cout<<"* Inserte n mero de filas para la matriz: ";
502     cin>>filas;
503     cout<<endl;
504     cout<<"* Inserte n mero de columnas para la matriz: ";
505     cin>>columnas;
506     cout<<endl;
507
508     if(filas==columnas)
509     {
510
511         //Inserta los valores para cada elemento de la matriz
512         cout<<" "<<endl;
513         cout <<"DATOS PARA LA MATRIZ: "<<endl;
514         int M[filas][columnas];
515
516         for(size_t i{0}; i<filas; i++)
517         {
518             for(size_t j{0}; j<columnas; j++)
519             {
520                 cout <<"Matriz["<<i<<"]["<<j<<"]: ";
521                 cin >> M[i][j];
522             }
523         }
524
525
526         //Imprime Matriz Original
527         cout<<" "<<endl;
528         cout << "Matriz Original";
529         cout<<" "<<endl;
530
531         for(size_t i{0};i<filas;i++)
532         {
533             cout <<" | ";
534             for(size_t j{0}; j<columnas; j++)
535             {
536                 cout<<M[i][j]<<" ";
537             }
538             cout << " | "<<endl;
539         }
540         cout<<" "<<endl;
541
542
543         //Imprime Matriz Transpuesta
544         cout<<" "<<endl;
545         cout << "Matriz Transpuesta";
546         cout<<" "<<endl;
547
548         for(size_t i{0};i<filas;i++)
549         {
550             cout <<" | ";

```

```

551     for(size_t j{0}; j<columnas; j++)
552     {
553         cout<<M[j][i]<<" ";
554     }
555     cout << " | "<<endl;
556 }
557 cout<<" "<<endl;
558 }
559
560 else
561 {
562     cout<<" "<<endl;
563     cout<<"                                AVISO          !!!!!"<<endl;
564     cout<<"-----" <<endl;
565     cout<<" "<<endl;
566     cout<<"                                NO SE PUEDE REALIZAR LA OPERACION"<<endl;
567     cout<<" "<<endl;
568     cout<<"                                EL N MERO DE COLUMNAS"<<endl;
569     cout<<"                                NO ES EL MISMO N MERO DE FILAS"<<endl;
570     cout<<"-----" <<endl;
571     cout<<" "<<endl;
572 }
573
574 }

```

VI. CONCLUSIONES

VII. BIBLIOGRAFÍA

- Grossman, S. I. (2008). Álgebra lineal. McGraw Hill Educación.