

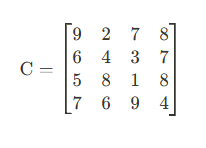
**UNVIME Escuela de Ingeniería y Ciencias Ambientales - Ingeniería en Sistemas**

**Materia: Análisis y Diseño de Algoritmos**

**Alumno: Gimenez Thomas Valentin**

**Desarrollo:**

El problema de la asignación por búsqueda exhaustiva se basa en la siguiente matriz de costos:



Lo que se busca es encontrar la permutación con el menor costo posible entre todas las permutaciones entre los 4 empleados y los 4 trabajos disponibles

Exponiendo un ejemplo de la primera permutación nos quedaría así:

⟨1,2,3,4⟩= Costo C[1,1]+C[2,4]+C[3,3]+C[4,4]= 9+4+1+4= 18

Entonces nos quedaría un en total un coste de 18 en la permutación ⟨1,2,3,4⟩

Hay que repetir el procedimiento para cada una de las permutaciones

Por suerte sabemos que en total hay n! permutaciones por lo que en este caso son 4! = 24 permutaciones posibles

Vistas en modo de cuadro quedarían las permutaciones tal que:

|  |  |
| --- | --- |
| **Permutación** | **Costo total** |
| **⟨1,2,3,4⟩**  **⟨1,2,4,3⟩**  **⟨1,3,2,4⟩**  **⟨1,3,4,2⟩**  **⟨1,4,2,3⟩**  **⟨1,4,3,2⟩**  **⟨2,1,3,4⟩**  **⟨2,1,4,3⟩**  **⟨2,3,1,4⟩**  **⟨2,3,4,1⟩**  **⟨2,4,1,3⟩**  **⟨2,4,3,1⟩**  **⟨3,1,2,4⟩**  **⟨3,1,4,2⟩**  **⟨3,2,1,4⟩**  **⟨3,2,4,1⟩**  **⟨3,4,1,2⟩**  **⟨3,4,2,1⟩**  **⟨4,1,2,3⟩**  **⟨4,1,3,2⟩**  **⟨4,2,1,3⟩**  **⟨4,2,3,1⟩**  **⟨4,3,1,2⟩**  **⟨4,3,2,1⟩** | **18**  **30**  **24**  **26**  **33**  **23**  **13**  **24**  **14**  **24**  **22**  **27**  **14**  **22**  **16**  **23**  **19**  **29**  **18**  **24**  **20**  **23**  **22**  **25** |

De esta manera sacamos que la permutación menos costosa es la **⟨2,1,3,4⟩** con coste **13**