Antes de generar la rotación:

- El responsable de genererar la rotación tiene que seleccionar las personas que van a realizar los puestos, de entre la lista de quienes están presentes, puede pasar que sobren operarios.

- El programa tiene que realizar un grupo con las personas que saben 4 operaciones o menos para asignarlas primero y luego a los demás.

-

**Pasos para generar la rotación:**

Preparacion de vector con operaciones aleatorias

* Cargar vector con las 26 operaciones en forma aleatoria en las posiciones del vector que se va a usar para asignar operaciones en orden.
* Seleccionar la primer posición del vector con las operaciones aleatorias, a la ultima.

Preparación de vector con los legajos cargados para rotar

* Armar un vector de 26 posiciones con los legajos de los operarios disponibles para la rotación.
* Separar de esos 26 legajos a los operarios que se saben 4 o menos operaciones
* Asignar primero a las rotaciones esos operarios

Agisnacion de operarios a operaciones

-con el grupo de operarios que saben menos operaciones ordenarlo de quien sabe menos operaciones a quien sabe mas y ubicarlos en los puestos.

-con el segundo grupo de operarios que sabe mas de 4 operaciones la asignación es inversa. Primero selecciono las operaciones que quedaron sin asignar y busco entre los operarios que restaron del vector operarios disponibles quien se sabe las operaciones.

Asignación bloque siguiente

-con el primer bloque cargado realizar los pasos anteriores se asignación, pero con cada asignación comparar con el bloque anterior que no sea la misma. En el caso de que los operarios solo sepan 2 o 3 operaciones si o si tendrá que repetir una operación, las personas que sepan 4 o mas operaciones no deben repetir operaciones

-si la asignación de operarios no fue exitosa deberá realizar nuevamente la asignación de operarios desde el principio y probar con otra combinación.

Objetivos del programa:

-Si el operario sabe 4 operaciones que no repita ninguna de las 4 en los 4 bloques.

-la rotación de la semana siguiente tiene que ser diferente a la de la anterior.

-cargar legajo nuevo el mismo no va poder modificarse, si se carga mal tendrá que eliminarse.

-editar legajo ya cargado. Los datos de mismo serán legajo, nombre y apellido, operaciones aprendidas.

-eliminar legajos.

-modificar disponibilidad para armar rotaciones

-modificar si operario esta activo (trabajando)

-listar operarios que están disponibles para las rotaciones .

-listar operarios activos

-editar cualquier propiedad a partir del legajo

-todas la propiedades excepto el legajo tiene que poder modificarse.

-las operaciones aprendidas tienen que poder agregarse como eliminarse

-al momento de armar la rotación el programa tiene que listar los operarios activos e ir seleccionando los que va a estar disponibles para la rotación, hasta tener los 26, cuando ya los tenga que no deje agregar mas operarios

-al momento de generar la rotación tiene que preguntar al usuario por una confirmación para poder guardarla y dar la oportunidad de poder generar otra.

-al seleccionar armar rotación que el programa verifique que tenga mínimo 26 operarios disponibles para las rotaciones.

-al agregar operarios para de rotación los legajos ingresados tiene que estar en la lista de disponibles y no repetirse

-la rotación se almacena en un objeto

-

clase

Datos empleado

Legajo int

Nombre y apellido char

Operaciones aprendidas vector[26]

Disponibilidad para agregar a la rotación bool

Operario activo bool

clase

Rotación

Rotacion [4] [26]

Fecha de la rotación