EJERCICIO PREPARACION DE CERTAMEN DAI

1.- Haciendo uso de tablas internas en un programa cobol, apoyar a la Administración de la Sede en el control de personal.

Se requiere almacenar los datos que identifican a cada funcionario, vale decir su rut y nombre. Se almacenara también la lista de Departamentos o área al que pertenecen. El total de funcionarios a manejar en la aplicación es un máximo de 100 personas y las áreas de la universidad son 30.

La tabla de funcionarios y de departamentos se efectuará en forma independiente, se recomienda al inicio del programa. En la tabla de funcionarios se debe considerar adicionalmente un campo que permita acumular los minutos de atrasos de cada funcionario.

move 1 to ContRut.

Display "desea ingresar datos?"

accept fin.

Perform until fin = "n"

Display "rut"

Accept pos

if rut[pos] not = 0

display "rut esta ocupado"

end-if

if rut[pos] = 0 and pos > 0 and pos < 101

add 1 to ContRut

move pos to rut[pos]

display "nombre"

accept nombre[pos]

else display "rut debe estar entre 0 y 100"

end-if

display "desea seguir?"

accept fin

if ContRut = 101

display "el registro esta lleno"

move "n" to fin

end-if

end-perform.

move 1 to ContArea

display "desea ingresar area?"

accept fin

perform until fin = "n"

display "area"

accept pos

if area[pos] not = 0

display "Ya existe"

else if area[pos] = 0 and pos > 0 and pos < 31

move pos to area[pos]

display "nombre"

accept nombre[pos]

add 1 to ContArea

else display "el valor debe estar entre 1 y 30"

end-if

display "desea continuar?"

accept fin

if ContArea = 31

Display "registro lleno"

move "n" to fin

end-if

end-perform.

SE recibirá por pantalla los datos de rut de funcionario y departamento los que deben ser almacenados en una matriz bidimensional en que el eje X corresponde al funcionario y el Y a los departamentos. En el elemento de la tabla debe ser registrado un “S” al ser recibida la combinación. Recordar que las tablas tienen un índice en que se podrá determinar la posición de la entidad buscada en la tabla.

display "desea ingresar combinaciones?"

accept fin

perform until fin = "n"

Display "rut"

accept pos-r

Display "area"

accept pos-a

if rut[por-r] not = 0 and area[pos-a] not = 0

move "s" to registro[pos-r,pos-a]

end-if

display "desea continuar"

accept fin

end-perform.

Se recibirá finalmente el rut del funcionario y el número de minutos de atraso, los que deben ser acumulados en la tabla de funcionarios.

display "desea ingresar atrasos?"

accept fin

perform until fin = "n"

display "rut"

accept pos-r

if rut[pos-r] > 0

display "ingrese minutos de atraso"

accept atrasos[pos-r]

else display "no existe el rut"

end-if

display "desea continuar?"

accept fin

end-perform.

2.- Efectuar los mismos procesos anteriores pero reemplazando las tablas internas por archivos. Definir las DDS que se usará en las definiciones de los archivos.

aqui no sabia si habia que mover las tablas a los archivos o hacer un nuevo registro, hice el 2do.

open i-o Funcionarios

display "desea ingresar datos?"

accept resp

move 1 to ContRut

move 1 to ContArea

perform until resp = "n"

Display "ingrese rut"

accept Rut

If Rut > 0 and Rut < 101

Read Funcionarios invalid key Display "ingrese Nombre"

accept nombre of funcionarios

move zeroes to atraso of F

write reg-f

add 1 to ContRut

not invalid key display "ya ingresado"

end-read

else Display "rut debe estar entre 0 y 100"

end-if

Display "Desea seguir ingresando datos? s/n"

accept resp

If ContRut = 101

move "n" to resp

display "registro lleno"

end-if

end-perform

open i-o Area

display "desea ingresar datos?"

accept resp

perform until resp = "n"

display "ingrese area"

accept area

if area of A < 31 and > 0

Read Areas invalid key display "ingrese nombre"

accept nombre of areas

write reg-A

add 1 to ContArea

not invalid key display "ya ingresado"

end-read

else display "area debe estar entre 0 y 30"

end-if

if i = 31

display "registro lleno"

move "n" to resp

end-if

end-perform

display "desea llenar la matriz?"

accept resp

perform until fin = "n"

move "s" to existe

display "rut"

accept rut

read funcionarios invalid key move "n" to existe

display "area"

accept area

read areas invalid key move "n" to existe

if existe = "s"

move rut to rutM

move area to areaM

Read Matriz invalid key move "s" to registro

write reg-M

not invalid key display "registro ya ingresado"

end-read

end-if

display "desea seguir ingresando datos?"

accept resp

end-perform

display "desea ingresar atrasos?"

accept resp

perform until resp = "n"

display "rut"

accept rut

if rut > 0 and < 101

Read Funcionarios invalid key display "rut no ingresado"

not invalid key display "ingrese atraso"

accept atraso

rewrite reg-F

end-read

else display "rut no valido"

end-if

display "desea seguir?"

accept resp

end-perform

3.- Para ambos casos, generar informes que permitan saber cuáles son los funcionarios que tienen la mayor y menor cantidad de atrasos.

hago un corrimiento ya que nose si con informe se refiere a mostrar 2 o 3 datos.

move 0 to i

perform until i = 99

add 1 to i

move 1 to j

perform until j = 99

if rut[j] < rut[j+1]

move rut[i] to aux

move rut[j] to rut[i]

move aux to rut[j]

end-if

add 1 to j

end-perform

end-perform.

Para mostrar el registro lo hubiera hecho con un campo donde se marcara el que muestro al pasar cada una de las veces, haria un display pasandole una variable max que empieze en 0 y despues una min en 999 y las cambio a 0 y 999 cada vez que paso, al terminar muevo ceros a estas variables para limpiarlas.

4.- Permitir consultar si un determinado funcionario trabaja o no para un determinado departamento indicados por pantalla.

display "rut"

display "dpto"

.1 accept pos-r

accept pos-a

If registro[pos-r,pos-a] = "s"

display "funcionario trabaja en este departamento o area"

else display "no trabaja en esta area"

end-if

.2 accept rutM

accept areaM

Read M invalid key display "datos erroneos"

not invalid key display "funcionario trabaja en este departamento o area"

end-read

close A

close F

close M

Stop run.