

```

#include <stdio.h>
#include <stdlib.h>
struct nodo
{
    int valor;
    //int dato;
    struct nodo *std;
};

struct celda
{
    int id;
    struct nodo *lista;
};

int agregar(celda a[100], int cant, celda nueva)    // int id
{
    a[cant]=nueva;
    cant++;
    return cant;
}

int existe(celda a[100], int cant, int id)
{
    int rta = -1;
    int i = 0;
    while (i<cant && rta == -1)
    {
        if (a[i].id == id)
        {
            rta = i;
        }
        i++;
    }
    return rta;
}

int agregartodo(celda a[100],int cant, int id, int valor)
{
    nodo *aux = crearnodo(valor);
    int i = existe(a, cant, id);
    if (i == -1)
    {
        celda una;
        una.id=id;
        una.lista=null;
        cant = agregar(a, cant, una);
        i = cant - 1;
    }
    a[i].lista = agregarppio(a[i].lista, aux);
    return cant;
}

```

```

nodo *buscar (celda a[100], int cant, int id, int valor)
{
    nodo *aux = null;
    int pos = existe(a, cant, id);
    if (pos != -1)
    {
        aux = buscarnodo(a[pos].lista, valor);
    }
    return aux;
}

```

```

main ()
{
    celda a[100];
}

```

```

struct registro
{
    mat;
    alu;
    nota;
};

```

```

int pasar (char nombre_archivo[30], celda a[100], int cant)
{
    file *archi=fopen (.....);
    registro unreg;
    while (!feof(archi))
    {
        fread(&unreg,.....);
        cant = agregartodo(a, cant, unreg.mat, unreg.nombre, unreg.nota);
    }
    fclose(archi);
    return cant;
}

void pasar_de_arreglo_a_archivo(char nombre[30],celda a[100], int cant)
{
    nodo *seg;
    registro unreg;
    file *archi = fopen(      );
    for (i = 0; i < cant; i++)
    {
        seg = a[i].lista;
        unreg.mat = a[i].mat;
        while (seg != null)
        {
            strcpy(unreg.alu, seg->alu);
            unreg.nota = seg->nota;
            fwrite(&unreg,.....);
            seg = seg->ste;
        }
    }
    fclose(archi);
}

```