

GUIA ESTRUCTURAS ENLAZADAS

```
/*Utilizando las funciones de ListasSE provistas en clase haga un programa que permita al usuario:
```

- Insertar un elemento ordenadamente en la lista
- Buscar el mayor elemento
- Calcular el promedio de la lista*/

```
struct NodoLista
{ int info;
  NodoLista *sgte; }

void insertarordenado (NodoLista *l, int num)
{ NodoLista* aux=l;
  NodoLista* ant=NULL;
  while (aux!=NULL && (num < aux->info)
  {
    NodoLista* nuevo = new NodoLista ();
    ant->sgte= nuevo;
    nuevo->info =x;
    nuevo -> sgte =aux; }
  ant= aux;
  aux= aux->sgte;}
new NodoLista ();
}

void mayorelemento (NodoLista *l)
{ NodoLista* aux=l;
  int mayor;
  while(aux!=NULL)
  { if (aux->info > mayor)
    {mayor= aux->info;}
    aux=aux->sgte;}}

float calcularpromedio (NodoLista *l)
{ int cant=0;
  int sum;
  NodoLista* aux=l;
  while(aux!=NULL)
  { suma= suma+ l->info;
    cant++;
    aux=aux->sgte;
  }
  return suma/cant;}

int main()
{}
```

```
/*Realice un programa que lea de un archivo datos de alumnos
de un curso. A medida que el archivo se va leyendo va insertando
los registros en una lista. Liste finalmente
```

```

todos los nodos para verificar que la operación fue exitosa.*/
//Cada alumno tiene una sublista de materias aprobadas

struct Alumno
{ char nombre[50];
  char apellido [50];
  char dni[10];
  int legajo;
  int codigocarrera;}

struct Materias
{char nombremat[50];
 char fechaaprobacion[11];
 int nota;
 int libro;
 int folio; }

struct Nodolista
{ Alumno info;
  Nodolista* sgte=NULL; }

void mostrarlista (Nodolista *l)
{ while(l)
  {cout << "===== DATOS DEL ALUMNO " << lista->info.apellido << "======" << endl;
   cout << "Apellido y nombre: " << lista->info.apellido << ", " << lista->info.nombre << endl;
   cout << "DNI: " << lista->info.dni << endl;
   cout << "Legajo: " << lista->info.legajo << endl;
   cout << "Carrera: " << lista->info.codcarrera<< endl << endl;
   l=l-> sgte;}}

void insertaralfinal(Nodolista* l, Alumno a)
{ Nodolista *nuevo = new Nodolista();
  nuevo -> info = a;
  nuevo -> sgte =NULL;
  if (l==NULL)
  {l=nuevo;}
  else
  {Nodo* aux=l;
   while(aux->sgte!=NULL)
   {aux=aux->sgte;}
   aux ->sgte=nuevo;}}

void pasaralista (Nodolista* l, FILE* f)
{ Alumno a;
  while (fread(&a,sizeof(Alumno),1,f))
  {insertaralfinal (l, a)}
}

int main()
{ f=fopen("Alumnos.dat", "rb");
  Nodolista* l= NULL;
  pasaralista(l,f);
  mostrarlista(l);}

```