

ALGORITMOS DE CLASIFICACIÓN INTERNA

	Pseudo-código
Tipos de datos	Const MAX=1000 Tipos tipo_vector: vector[1..MAX] de tipo_elemento tipo_elemento: registro de clave: tipo_clave {otros campos} Fin_registro tipo_clave: entero
Inserción Directa	Procedimiento inserción_directa(a: tipo_vector(E/S), n: entero(E)) Var i,j: entero x: tipo_elemento Inicio Desde i=2 Hasta n Hacer x ← a[i] j ← i-1 Mientras (j ≥ 1) y (x.clave < a[j].clave) Hacer a[j+1] ← a[j] j ← j-1 Fin_mientras a[j+1] ← x Fin_desde Fin_procedimiento
Selección Directa	Procedimiento SelecciónDirecta(a: tipo_vector (E/S), n: entero(E)) Var i,j,k: entero x: tipo_elemento Inicio Desde i=1 Hasta n-1 Hacer k ← i Desde j=i+1 Hasta n Hacer Si (a[j].clave < a[k].clave) Entonces k ← j Fin_si Fin_desde x ← a[k] a[k] ← a[i] a[i] ← x Fin_desde Fin_procedimiento
Intercambio Directo (Burbuja)	Procedimiento Burbuja(a: tipo_vector (E/S), n: entero (E)) Var i,j: entero x: tipo_elemento Inicio Desde i=2 Hasta n Hacer Desde j=n Hasta i paso=-1 Hacer Si (a[j].clave < a[j-1].clave) Entonces x ← a[j] a[j] ← a[j-1] a[j-1] ← x Fin_si Fin_desde Fin_desde Fin_procedimiento

ALGORITMOS DE BÚSQUEDA BINARIA

	Pseudo-código
Tipos de datos	Const MAX=1000 Tipos tipo_vector: vector[1..MAX] de tipo_elemento tipo_elemento: registro de clave: tipo_clave {otros campos} Fin_registro tipo_clave: entero
Primera ocurrencia de clave (primer elemento de clave mayor o igual)	Procedimiento Búsqueda_binaria(a: tipo_vector (E/S), n: entero (E), clave:tipo_clave(E), encontrado: lógico (S), pos: entero (S)) Var izq,der,med: entero Inicio izq←1 der←n Mientras (izq≤der) Hacer med←(izq+der) DIV 2 Si (clave≤a[med].clave) Entonces der←med-1 Sino izq←med+1 Fin_si Fin_mientras pos←izq encontrado←(izq≤n) y (a[izq].clave=clave) Fin_procedimiento
Posición de inserción (primer elemento de clave estrictamente mayor)	Funcion Búsqueda_pos_insercion(a: tipo_vector (E/S), n: entero (E), clave:tipo_clave(E)): entero Var izq,der,med: entero Inicio izq←1 der←n Mientras (izq≤der) Hacer med←(izq+der) DIV 2 Si (clave<a[med].clave) Entonces der←med-1 Sino izq←med+1 Fin_si Fin_mientras Devolver (izq) Fin_función

ALGORITMOS DE CLASIFICACIÓN INTERNA

	Lenguaje de programación C
Tipos de datos	<pre>#define MAX 1000 typedef struct{ int clave; /*resto de componentes*/ } tipo_elemento; typedef tipo_elemento tipo_vector[MAX];</pre>
Inserción Directa	<pre>void insercion(tipo_vector a,int n){ int i,j; tipo_elemento x; for (i=1;i<n;++i){ x=a[i]; j=i-1; while ((j>=0)&&(x.clave<a[j].clave)){ a[j+1]=a[j]; j=j-1; } a[j+1]=x; } }</pre>
Selección Directa	<pre>void seleccion(tipo_vector a,int n){ int i,j; tipo_elemento x; int k; for (i=0;i<n-1;++i){ k=i; for (j=i+1;j<n;++j) if (a[j].clave < a[k].clave) k=j; x=a[k]; a[k]=a[i]; a[i]=x; } }</pre>
Intercambio Directo (Burbuja)	<pre>void burbuja(tipo_vector a,int n){ int i,j; tipo_elemento x; for (i=1;i<n;i++) for (j=n-1;j>=i;j--) if (a[j].clave<a[j-1].clave){ x=a[j]; a[j]=a[j-1]; a[j-1]=x; } }</pre>

ALGORITMOS DE BÚSQUEDA BINARIA

	Lenguaje de programación C
Tipos de datos	<pre>#define MAX 1000 typedef struct{ int clave; /*resto de componentes*/ } tipo_elemento; typedef tipo_elemento tipo_vector[MAX];</pre>
Primera ocurrencia de clave (primer elemento de clave mayor o igual)	<pre>void Busqueda_binaria(tipo_vector a, int n, int clave, int *encontrado, int *pos){ int izq,der,med; izq=0; der=n-1; while(izq<=der){ med=(izq+der)/2; if(clave<=a[med].clave) der=med-1; else izq=med+1; } *pos=izq; *encontrado←(izq<=n)&&a[izq].clave==clave); }</pre>
Posición de inserción (primer elemento de clave estrictamente mayor)	<pre>int Busqueda_pos_insercion(tipo_vector a, int n, int clave){ int izq,der,med; izq=0; der=n-1; while(izq<=der){ med=(izq+der)/2; if(clave<a[med].clave) der=med-1; else izq=med+1; } return (izq); }</pre>