

Fundamentos de Organización de Datos

1

Curso 2017

Fundamentos de Organización de Datos

Archivos
Longitud Variable

Campos y registros con longitud variable

Ejemplo: Algoritmo de creación de un archivo de empleados. De los mismos se conoce nombre, apellido y dirección. Cada registro debe ser almacenado maximizando el aprovechamiento de espacio.

Campos y registros con longitud variable

- ✓ El archivo debe definirse de tipo **file**.
Permite realizar la transferencia de la información carácter a carácter.
- ✓ Marca de fin de campo: Se utilizará el carácter **#**.
- ✓ Marca de fin de registro: Se utilizará el carácter **@**.

```
program ejemplo;
```

```
var
```

```
    empleados: file; {archivo sin tipo}
```

```
    nombre, apellido, direccion: string;
```

```
    documento: word;
```

```
begin
```

```
    assign (empleados, 'empleados.dat');
```

```
    rewrite (empleados, 1);
```

```
    writeln('Ingrese el apellido');
```

```
    readln(apellido);
```

```
    while (apellido <> ' ') do begin
```

```
        writeln('Ingrese el nombre');
```

```
        readln(nombre);
```

```
        writeln('Ingrese la direccion');
```

```
        readln(direccion);
```

Tamaño en bytes
de los bloques
que se van a usar
para la escritura

Cantidad de bloques a escribir en el archivo

```
BlockWrite (empleados, apellido, length (apellido) +1);  
BlockWrite (empleados, '#', 1);  
BlockWrite (empleados, nombre, length (nombre) +1);  
BlockWrite (empleados, '#', 1);  
BlockWrite (empleados, direccion, length (direccion) +1);  
BlockWrite (empleados, '@', 1);  
  
writeln('Ingrese el apellido');  
readln (apellido);  
end; {while}  
close (empleados);  
end.
```

Para variables numéricas se debe utilizar **SizeOf()**

Ejemplo

El siguiente algoritmo permite recorrer el archivo anteriormente generado y presenta los datos en pantalla.

Ejemplo

```
program ejemplo;
```

```
var
```

```
empleados: file; {archivo sin tipo}
```

```
buffer, campo: string;
```

```
begin
```

```
assign (empleados, 'empleados.dat');
```

```
reset (empleados, 1);
```

**Tamaño en bytes
de los bloques a
usar para la lectura**


```
while not (EOF(empleados)) do begin
  blockread(empleados, buffer, 1);
  while (buffer <> '@') and not (EOF(empleados)) do begin
    campo := '';
    while (buffer <> '@') and (buffer <> '#')
      and not (EOF(empleados)) do begin
      campo := campo + buffer ;
      blockread(empleados, buffer, 1);
    end;
    writeln(campo);
    if not (EOF(empleados)) then
      blockread(empleados, buffer, 1);
  end;
end;
close(empleados);
```