Archivos

Computer Science

CS1100 - Introducción a Ciencia de la Computación

UTEC
UNIVERSIDAD DE INGENIERÍA
Y TECNOLOGÍA

Archivos

Dispositivos de almacenamiento de Archivos

Logro de la Sesión

Al finalizar esta sesión, el alumno será capaz de:

- Leer archivos de texto.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Logro de la Sesión

Al finalizar esta sesión, el alumno será capaz de:

- Leer archivos de texto.
- Generar archivos de texto.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Logro de la Sesión

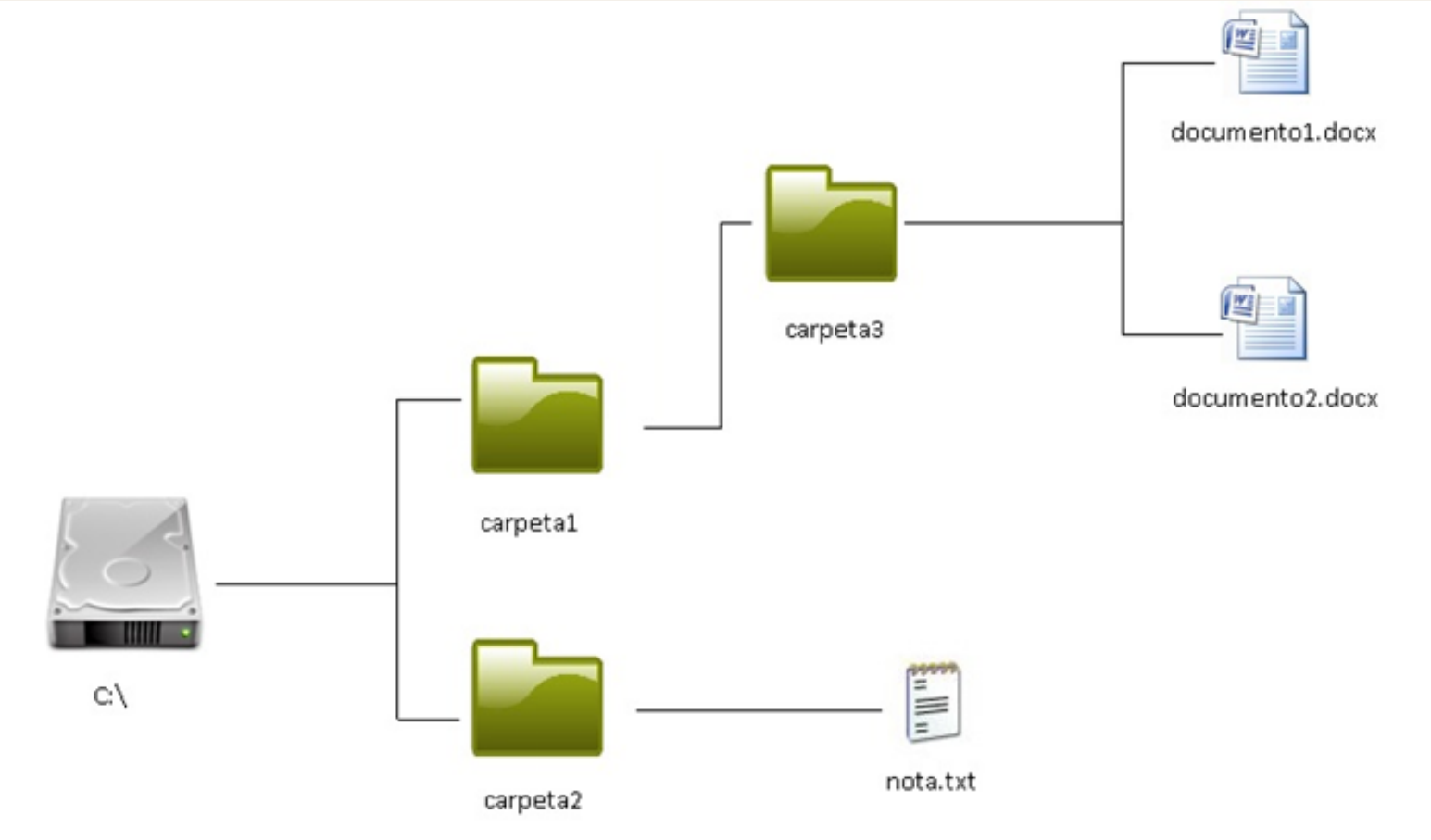
Al finalizar esta sesión, el alumno será capaz de:

- Leer archivos de texto.
- Generar archivos de texto.
- Trabajar con archivos de texto.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Sistema de Archivos

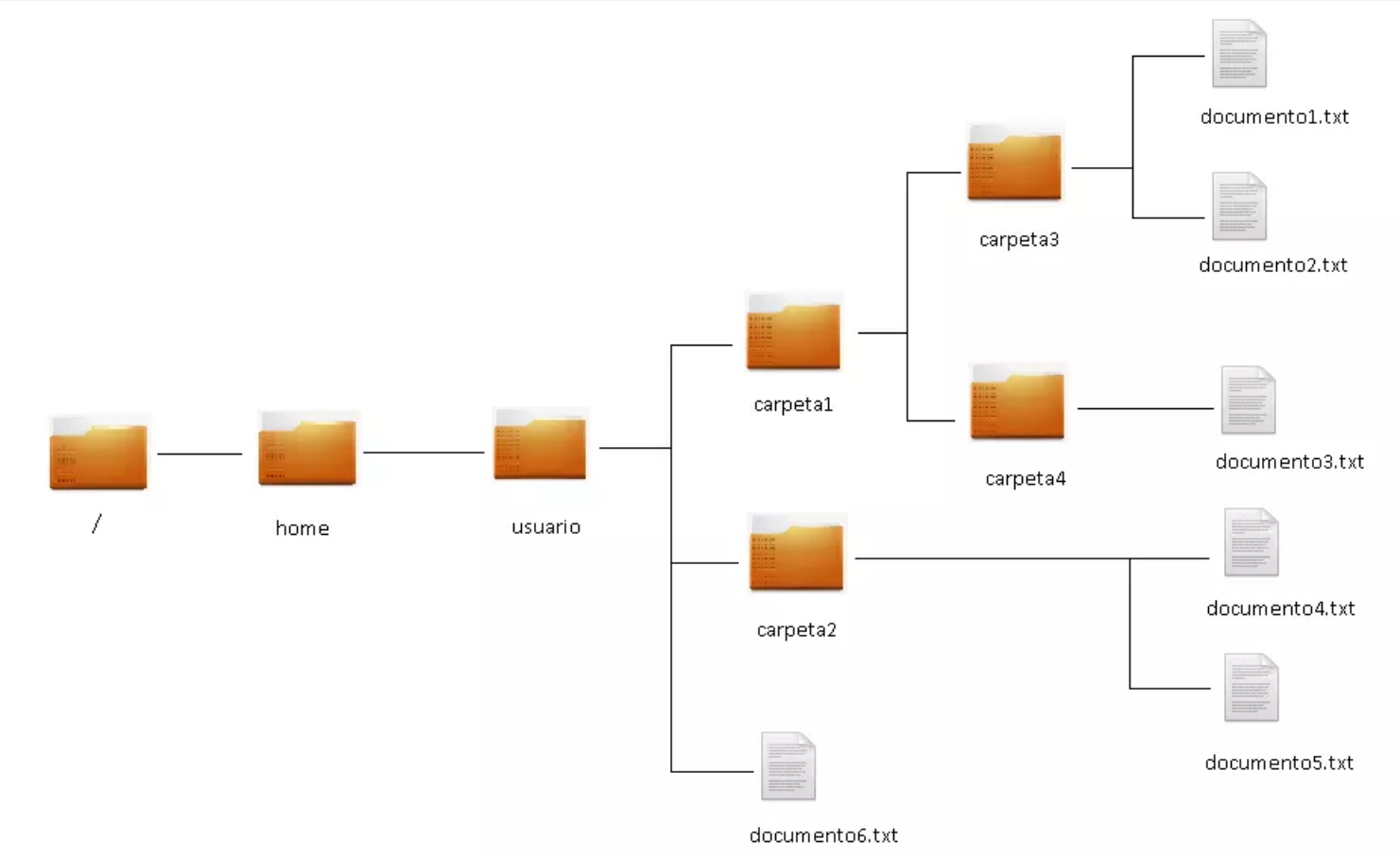
Windows



UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Sistema de Archivos

Linux



Definicíon

£Qué es una Archivo?

- También llamados ficheros, son en resumidas cuentas, un conjunto de bytes almacenados en un dispositivo.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Definicíon

£Qué es una Archivo?

- También llamados ficheros, son en resumidas cuentas, un conjunto de bytes almacenados en un dispositivo.
- Este dispositivo suele ser el disco duro de la PC, pero también podría ser un CD. Incluso en nuestros celulares también tenemos archivos.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Definicíon

£Qué es una Archivo?

- También llamados ficheros, son en resumidas cuentas, un conjunto de bytes almacenados en un dispositivo.
- Este dispositivo suele ser el disco duro de la PC, pero también podría ser un CD. Incluso en nuestros celulares también tenemos archivos.
- Ejemplos comunes de archivos son: Documentos hechos en word o excel (doc, xls). Archivos de canciones (mp3). Archivos de video (mp4), Archivos de texto (txt), etc.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Definicíon

£Qué es una Archivo?

- También llamados ficheros, son en resumidas cuentas, un conjunto de bytes almacenados en un dispositivo.
- Este dispositivo suele ser el disco duro de la PC, pero también podría ser un CD. Incluso en nuestros celulares también tenemos archivos.
- Ejemplos comunes de archivos son: Documentos hechos en word o excel (doc, xls). Archivos de canciones (mp3). Archivos de video (mp4), Archivos de texto (txt), etc.
- Sobreviven aún cuando se apague la PC.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Abriendo un archivo

Para abrir un archivo se usa la función **open()**:

```
1 <objeto_archivo> = open(<archivo>, <modo>)
```

Abre el `<archivo>` y devuelva un `<objeto_archivo>`. Si el `<archivo` no se puede abrir, se genera un error **OSError**.

- `<archivo>` es un objeto que da la ruta de acceso (absoluta o relativa al directorio de trabajo actual) del archivo que se abrirá o un descriptor de archivo.
- `<modo>` es una cadena opcional que especifica el modo en el que se abre el archivo. El valor predeterminado es **'r'**, que significa abierto para leer en modo texto. **'w'** para escritura (truncando el

Abriendo un archivo

Para abrir un archivo se usa la función **open()**:

```
1 <objeto_archivo> = open(<archivo>, <modo>)
```

Abre el `<archivo>` y devuelva un `<objeto_archivo>`. Si el `<archivo` no se puede abrir, se genera un error **OSError**.

- `<archivo>` es un objeto que da la ruta de acceso (absoluta o relativa al directorio de trabajo actual) del archivo que se abrirá o un descriptor de archivo.
- `<modo>` es una cadena opcional que especifica el modo en el que se abre el archivo. El valor predeterminado es **'r'**, que significa abierto para leer en modo texto. **'w'** para escritura (truncando el

Abriendo un archivo

Para abrir un archivo se usa la función **open()**:

```
1 <objeto_archivo> = open(<archivo>, <modo>)
```

Abre el `<archivo>` y devuelva un `<objeto_archivo>`. Si el `<archivo` no se puede abrir, se genera un error **OSError**.

- `<archivo>` es un objeto que da la ruta de acceso (absoluta o relativa al directorio de trabajo actual) del archivo que se abrirá o un descriptor de archivo.
- `<modo>` es una cadena opcional que especifica el modo en el que se abre el archivo. El valor predeterminado es **'r'**, que significa abierto para leer en modo texto. **'w'** para escritura (truncando el

Abriendo un archivo

Modos de apertura: Parte 1

Indicador	Modo de apertura	Ubicación del puntero
`r`	Solo lectura	Al inicio del archivo
`rb`	Solo lectura en modo binario	Al inicio del archivo
`r+`	Lectura y escritura	Al inicio del archivo
`rb+`	Lectura y escritura en modo binario	Al inicio del archivo
`w`	Solo escritura. Sobreescribe el archivo si existe. Crea el archivo si no existe	Al inicio del archivo
`wb`	Solo escritura en modo binario. Sobreescribe el archivo si existe. Crea el archivo si no existe	Al inicio del archivo
`w+`	Escritura y lectura. Sobreescribe el archivo si existe. Crea el archivo si no existe	Al inicio del archivo

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Abriendo un archivo

Modos de apertura: Parte 2

`wb+`	Escritura y lectura en modo binario. Sobreescribe el archivo si existe. Crea el archivo si no existe	Al inicio del archivo
`a`	Añadido (agregar contenido). Crea el archivo si éste no existe	Si el archivo existe, al final de éste. Si el archivo no existe, al comienzo
`ab`	Añadido en modo binario (agregar contenido). Crea el archivo si éste no existe	Si el archivo existe, al final de éste. Si el archivo no existe, al comienzo
`a+`	Añadido (agregar contenido) y lectura. Crea el archivo si éste no existe.	Si el archivo existe, al final de éste. Si el archivo no existe, al comienzo
`ab+`	Añadido (agregar contenido) y lectura en modo binario. Crea el archivo si éste no existe	Si el archivo existe, al final de éste. Si el archivo no existe, al comienzo

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Leyendo datos de un archivo: for

Para leer datos de un archivo se utiliza la sentencia **for**:

```
1 for <cadena_linea> in <objeto_archivo>:
2     <sentencias>
```

- La lectura se realiza dentro del cuerpo del bucle.
- Cada ejecución del bucle leerá una línea del archivo en una cadena.

Ejemplo: Imprime el contenido del archivo

```
1 inputFile = open("datos.txt", "r")
2 for cadena in inputFile
3     print(cadena)
```

Leyendo datos de un archivo: for

Para leer datos de un archivo se utiliza la sentencia **for**:

```
1 for <cadena_linea> in <objeto_archivo>:
2     <sentencias>
```

- La lectura se realiza dentro del cuerpo del bucle.
- Cada ejecución del bucle leerá una línea del archivo en una cadena.

Ejemplo: Imprime el contenido del archivo

```
1 inputFile = open("datos.txt", "r")
2 for cadena in inputFile
3     print(cadena)
```

Leyendo datos de un archivo: readline()

Para leer datos de un archivo también se utiliza la función **readline()**:

```
1 <cadena_linea> = <objeto_archivo>.readline
    ()
```

- El método **readline()** permite obtener una línea de texto de un archivo.
- Un archivo que se encuentre abierto tiene una posición asociada, que indica el último punto que fue leído.
- Cada vez que se lee una línea, la posición de lectura avanza. Es por ello que readline() devuelve cada vez una línea distinta y no siempre la misma.

Ejemplo: Imprime el contenido del

Leyendo datos de un archivo: readline()

Para leer datos de un archivo también se utiliza la función **readline()**:

```
1 <cadena_linea> = <objeto_archivo>.readline
    ()
```

- El método **readline()** permite obtener una línea de texto de un archivo.
- Un archivo que se encuentre abierto tiene una posición asociada, que indica el último punto que fue leído.
- Cada vez que se lee una línea, la posición de lectura avanza. Es por ello que readline() devuelve cada vez una línea distinta y no siempre la misma.

Ejemplo: Imprime el contenido del

Leyendo datos de un archivo: readlines()

Para leer datos de un archivo también se utiliza la función **readlines()**:

```
1 <lista_cadena> = <objeto_archivo>.readlines
    ()
```

- El método **readlines()** permite obtener todas las líneas de un archivo.
- La función devuelva una lista, en donde cada elemento de la lista es una línea de texto del archivo.

Ejemplo: Imprime el contenido del archivo

```
1 inputFile = open("datos.txt", "r")
```

Leyendo datos de un archivo: readlines()

Para leer datos de un archivo también se utiliza la función **readlines()**:

```
1 <lista_cadena> = <objeto_archivo>.readlines
    ()
```

- El método **readlines()** permite obtener todas las líneas de un archivo.
- La función devuelva una lista, en donde cada elemento de la lista es una línea de texto del archivo.

Ejemplo: Imprime el contenido del archivo

```
1 inputFile = open("datos.txt", "r")
```

Cerrando un archivo: close()

Para cerrar un archivo se utiliza la función **close()**:

```
1 <objeto_archivo>.close()
```

- Aunque un archivo se cierra automáticamente cuando finaliza un programa, sigue siendo una buena práctica cerrarlo cuando el programa finalice.
- **£Qué sucede si un programa tiene un error en tiempo de ejecución y se bloquea antes cerrar los archivos abierto?** Los archivos puedes permanecer **bloqueados**, y pasar a un estado inaccesible, porque todavía está abierto.

Ejemplo: Imprime el contenido y cierra el archivo

Cerrando un archivo: close()

Para cerrar un archivo se utiliza la función **close()**:

```
1 <objeto_archivo>.close()
```

- Aunque un archivo se cierra automáticamente cuando finaliza un programa, sigue siendo una buena práctica cerrarlo cuando el programa finalice.
- **£Qué sucede si un programa tiene un error en tiempo de ejecución y se bloquea antes cerrar los archivos abierto?** Los archivos puedes permanecer **bloqueados**, y pasar a un estado inaccesible, porque todavía está abierto.

Ejemplo: Imprime el contenido y cierra el archivo

Ejemplo 1: Archivo de empleados

Desarrolle un programa que lea los datos del archivo **empleados.txt**, que tiene el siguiente contenido:

```
1 Juan Perez,CEO,3000
2 Morris Lee,CIO,4500
3 Jorge Rosas,CTO,10251
```

Y debe de imprimir los datos como se muestra a continuación:

```
1 Nombres: Perez, Juan Ocupacion: CEO Salario
    : $3000.00
2 Nombres: Lee, Morris Ocupacion: CIO Salario
    : $4500.00
```

Ejemplo 1: Archivo de empleados

La solución al ejemplo es:

```
1 inputFile = open("empleados.txt", "r")
2 for cadena in inputFile:
3     nombres, ocupacion, salariot = cadena.
        split(',')
4     nombre, apellido = nombres.split()
5     salario = int(salariot)
6     print("Nombres: %s, %s\tOcupacion: %s\
        tSalario: $%.2f" % (apellido, nombre
        , ocupacion, salario))
7 inputFile.close()
```

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Escribiendo en un archivo: w+ y write()

Para escribir en un archivo se utiliza la función **write()**:

```
1 <objeto_archivo>.write(<cadena>)
```

- Se recomienda usar el modo **w+** en la función **open()**, porque permite abrir el archivo en modo lectura y escritura.
- Con el modo **w+**, se escribe una cadena al principio del archivo.
- Usar la función **write()** junto con el `<objeto_archivo>`.
- Tenga en cuenta que esta función SÓLO acepta como parámetro una cadena (todo lo demás debe convertirse a este tipo).

Ejemplo: Escribiendo información del curso en el archivo datos.txt

```
1 inputFile = open("datos.txt", "w+")
```

Escribiendo en un archivo: w+ y write()

Para escribir en un archivo se utiliza la función **write()**:

```
1 <objeto_archivo>.write(<cadena>)
```

- Se recomienda usar el modo **w+** en la función **open()**, porque permite abrir el archivo en modo lectura y escritura.
- Con el modo **w+**, se escribe una cadena al principio del archivo.
- Usar la función **write()** junto con el `<objeto_archivo>`.
- Tenga en cuenta que esta función SÓLO acepta como parámetro una cadena (todo lo demás debe convertirse a este tipo).

Ejemplo: Escribiendo información del curso en el archivo datos.txt

```
1 inputFile = open("datos.txt", "w+")
```

Escribiendo en un archivo: a y write()

- Se utiliza el modo **a** en la función **open()** para escribir una cadena al final de un archivo.
- Usar la función **write()** junto con el `<objeto_archivo>`.

Ejemplo: Escribiendo información del curso en el final del archivo datos.txt

```
1 inputFile = open("datos.txt", "a")
2 inputFile.write("Finalmente\n")
3 inputFile.write("Bienvenidos al curso de
    ICC\n")
4 inputFile.write("del 2019-1\n")
5 inputFile.close()
```

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Escribiendo en un archivo: a y write()

- Se utiliza el modo **a** en la función **open()** para escribir una cadena al final de un archivo.
- Usar la función **write()** junto con el `<objeto_archivo>`.

Ejemplo: Escribiendo información del curso en el final del archivo datos.txt

```
1 inputFile = open("datos.txt", "a")
2 inputFile.write("Finalmente\n")
3 inputFile.write("Bienvenidos al curso de
    ICC\n")
4 inputFile.write("del 2019-1\n")
5 inputFile.close()
```

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Ejercicio 1

Desarrollar un programa en python que realice lo siguiente:

- Lea del teclado un número **n** y el nombre de un archivo.
- Cree el archivo con el nombre leido.
- Escriba en el archivo los número pares, desde cero hasta el número par menor o igual que **n**
- cierre el archivo.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Ejercicio 2

Desarrollar un programa en python que realice lo siguiente:

- Abra el archivo creado en el ejercicio 1.
- Cree el archivo copia.txt.
- Copie todo el contenido del archivo del ejercicio 1 al archivo copia.txt
- Cierre los archivos abiertos

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Ejercicio 3

Desarrolle programas en Python que ejecuten Insertion Sort sobre todas las permutaciones que generan las siguientes secuencias:

- *lista* = *random.sample*(*range*(10), 10)
- *lista* = [*random.randint*($i + 10$, $i +$ 100) *for i in range*(*random.randint*(10, 20))]
- Leer el archivo `"texto.txt"`: La Universidad Brinda Infraestructura y Educacion de Calidad para todos los Estudiantes
- *lista* = [*random.random*() *for_ in range*(20)]

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Ejercicio 4

Dado un diccionario de 20 elementos con llave entero y valor cadena, se pide ordenarlo por llave y por valor usando Insertion Sort, escogiendo el indice del menor valor de la derecha y agregandolo a la izquierda.

```
1 import random
2 import string
3 letters= string.ascii_lowercase
4 d = {key: value for key, value in zip(
    random.sample(range(300), 20), [""
    join(random.choice(letters) for i in
    range(n)) for n in random.sample(
    range(20, 40), 20)])}
```

Ejercicio 5

Dado un conjunto de numeros leidos del archivo: "numeros.txt"

```
1 14 25
2 23 12
3 45 56
4 23 45
5 12 34
6 90 12
```

Se pide ordenar los numeros de mayor a menor usando Insertion Sort.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Ejercicio 6

Del archivo poema.txt:

```
1 Caminemos en la era de la modernidad
2 Observando lo que ella nos puede dar
3 Manejar una computadora por empezar
4 Pulsando el ratón y el teclado virtual
5 Una manzanita y varios niños te guiaran
6 También una computadora sensacional
7 Así paso a paso vamos a aprender
```

Desarrollar un programa en python que realice lo siguiente:

- Crear el archivo palabras.txt en modo escritura.
- Abrir el archivo poema.txt en modo lectura.
- Contar todas las veces que aparece una palabra en el archivo poema.txt, y guarde la palabra y las veces que aparece en el

Cierre

En esta sesión aprendiste:

- Cómo abrir un archivo para leer.
- Cómo abrir un archivo para escribir.
- Los detalles de cómo se lee y escribe la información en un archivo.
- Cómo cerrar un archivo y por qué es una buena práctica hacerlo.
- Cómo leer desde datos de un archivo de tamaño arbitrario.
- Almacenamiento y procesamiento de datos usando archivos y funciones de cadena.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Cierre

En esta sesión aprendiste:

- Cómo abrir un archivo para leer.
- Cómo abrir un archivo para escribir.
- Los detalles de cómo se lee y escribe la información en un archivo.
- Cómo cerrar un archivo y por qué es una buena práctica hacerlo.
- Cómo leer desde datos de un archivo de tamaño arbitrario.
- Almacenamiento y procesamiento de datos usando archivos y funciones de cadena.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Cierre

En esta sesión aprendiste:

- Cómo abrir un archivo para leer.
- Cómo abrir un archivo para escribir.
- Los detalles de cómo se lee y escribe la información en un archivo.
- Cómo cerrar un archivo y por qué es una buena práctica hacerlo.
- Cómo leer desde datos de un archivo de tamaño arbitrario.
- Almacenamiento y procesamiento de datos usando archivos y funciones de cadena.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Cierre

En esta sesión aprendiste:

- Cómo abrir un archivo para leer.
- Cómo abrir un archivo para escribir.
- Los detalles de cómo se lee y escribe la información en un archivo.
- Cómo cerrar un archivo y por qué es una buena práctica hacerlo.
- Cómo leer desde datos de un archivo de tamaño arbitrario.
- Almacenamiento y procesamiento de datos usando archivos y funciones de cadena.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Cierre

En esta sesión aprendiste:

- Cómo abrir un archivo para leer.
- Cómo abrir un archivo para escribir.
- Los detalles de cómo se lee y escribe la información en un archivo.
- Cómo cerrar un archivo y por qué es una buena práctica hacerlo.
- Cómo leer desde datos de un archivo de tamaño arbitrario.
- Almacenamiento y procesamiento de datos usando archivos y funciones de cadena.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Cierre

En esta sesión aprendiste:

- Cómo abrir un archivo para leer.
- Cómo abrir un archivo para escribir.
- Los detalles de cómo se lee y escribe la información en un archivo.
- Cómo cerrar un archivo y por qué es una buena práctica hacerlo.
- Cómo leer desde datos de un archivo de tamaño arbitrario.
- Almacenamiento y procesamiento de datos usando archivos y funciones de cadena.

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA