

Práctica No. 13

Aplicaciones a nivel de bits

Objetivo: Desarrollar aplicaciones de software haciendo uso eficiente de la memoria.

Material:

- Computadora Personal (PC)
- Programa Editor de texto (ASCII), compilador GCC

Equipo:

- Computadora Personal

Desarrollo de la práctica:

Actividad 1.

Desarrollar una biblioteca para la manipulación de bits. Realizar un archivo .c donde se pruebe cada una de las funcionalidades programadas en la biblioteca para manipulación de bits.

Las funciones a implementar deberán ser las siguientes:

void display(unsigned int valor)

Función que imprime el valor de un número de forma binaria.

void clearbit(unsigned int valor, unsigned char pos)

Función que establece el valor 0 en un determinado bit (en la posición **pos**) de un número entero (**valor**).

void setbit(unsigned int valor, unsigned char pos)

Función que establece el valor 1 en un determinado bit (en la posición **pos**) de un número entero (**valor**).

unsigned char getbit(unsigned int valor, unsigned char pos)

Función que regresa el valor de un determinado bit (en la posición **pos**) de un número entero (**valor**).

unsigned int togglebit(unsigned int valor, unsigned char pos)

Función que intercambia el valor de un determinado bit (en la posición **pos**) de un número entero (**valor**)

unsigned int introtright(unsigned int valor, unsigned loc)

Función que rota a la derecha un determinado número de locaciones (definido por **loc**) de un número entero (**valor**)

unsigned introtleft(unsigned int valor, int loc)

Función que rota a la izquierda un determinado número de locaciones (definido por **loc**) de un número entero (**valor**)

unsigned char byterotright(unsigned char valor, unsigned loc)

Función que rota a la derecha un determinado número de locaciones (definido por **loc**) de un byte (**valor**)

unsigned char byterotleft(unsigned int char, int loc)

Función que rota a la izquierda un determinado número de locaciones (definido por **loc**) de un byte (**valor**)

unsigned char getbyte(unsigned int valor, unsigned char pos)

Función que regresa el contenido de un byte contenido en un número entero (**valor**) a partir de una posición otorgada por el usuario a través de la variable **pos**, la posición debe considerarse a nivel de bits.

void encrypt(char* mensaje, unsigned char clave)

Función que recibe un mensaje y aplica una máscara xor a cada caracter con la variable clave. De tal forma que se modifica el mensaje original por el encriptando.

Actividad 2.

Realizar un programa (ej. ultimateEncrypt.exe) que lea archivos de forma binaria y encripte su contenido con una clave. Hacer uso de la función **encrypt** desarrollada anteriormente. El programa debe recibir tres parámetros por consola:

- Nombre archivo que se ve a encriptar
- Segundo deberá ser una clave (número de 8 bits).
- Nombre del archivo de salida (archivo encriptado).

El tercer parámetro deberá ser opcional, en caso que se omita, se deberá guardar la información encriptado en un archivo con el mismo nombre del original pero agregándole la terminación “.crypt”.

El programa debe de imprimir la duración de su ejecución. Se debe considerar la encripción de archivos de cualquier tamaño.

Ejemplo de aplicación

```
> ultimateEncrypt.exe archivo.txt 148 encriptado.txt
```

EXTRA 10pts**unsigned int add(unsigned int num1, unsigned int num2)**

Función que realiza la suma de dos números empleando únicamente operaciones a nivel de bits.

unsigned int complemento2(unsigned int valor)

Función que realiza el complemento a dos de un número entero.

unsigned int sub(unsigned int num1, unsigned int num2)

Función que realiza la resta de dos números empleando únicamente operaciones a nivel de bits.

Restricciones

1. En un archivo .c diferente se deben de probar cada una de las funciones desarrolladas.
2. El código fuente deberá contener comentarios que indique el objetivo del programa y descripción de instrucciones más relevantes.
3. Evitar nombres de variables y funciones que no reflejen su propósito claramente.

Conclusiones y Comentarios.