

Laboratorio Avanzado
Javier de León
201603068

Tarea # 1

1. Compile, corra y analice el funcionamiento de los códigos:

Resultado de correr sizes.cpp:

- Se crean variables de 5 tipos diferentes y se muestra en consola el espacio que ocupan estas en memoria. Las variables que se crean son una bool y un char que ocupan 1 byte, un signed y unsigned int que ocupan 4 bytes y un float que ocupa también 4.

Resultado de correr inter.cpp:

- Se muestra la codificación en binario, hexa y decimal de un int y estas son guardadas como distintas variables, el número es signed y negativo.
- Se muestra el mismo resultado pero con un unsigned int y esto hace que la magnitud de este valor sea más grande.
- Se muestra que la notación científica no cambia el output de la codificación signed en binario.
- Se muestra el resultado de evaluar un signed int como bool, en este caso un valor de 0 es falso y cualquier otro resultado verdadero.
- Se codifica el número anterior como un float y se guarda en una variable distinta y se muestra como la codificación de notación científica de la cadena binaria.

2. Representación a 32 bits de signed int con complemento a 2:

Para encontrar la representación a 32 bits unsigned int de la magnitud del número hacemos aritmética con las potencias de 2. Luego para encontrar el complemento a 2 realizamos una inversión de los bits de la representación 32 bits unsigned y debemos realizar un cambio de bit de la componente más pequeña, esto es equivalente a la inversión de los bits +1. Realizaremos el ejemplo con el primer caso y luego todos se realizan de la misma manera.

- Primero caso, -125:

$$125 = 64+32+16+8+4+1$$

$$125 = 0....01111101$$

$$\text{Inversión (125)} = 1.....10000010$$

$$\text{Inversión (125)} + 1 = 1.....10000011$$

$$-125 = 11111111 \ 11111111 \ 11111111 \ 10000011$$

- Segundo caso, -4096:

$$4096 = 1000000000000$$

$$\text{Inversión (4096)} + 1 = -4096 = 11111111 \ 11111111 \ 11110000 \ 00000000$$

- Tercer caso, -1000000:

$$1000000 = 11110100001001000000$$

$$\text{Inversión(1000000)} + 1 = -1000000 = 11111111 \ 11110000 \ 10111101 \ 11000000$$