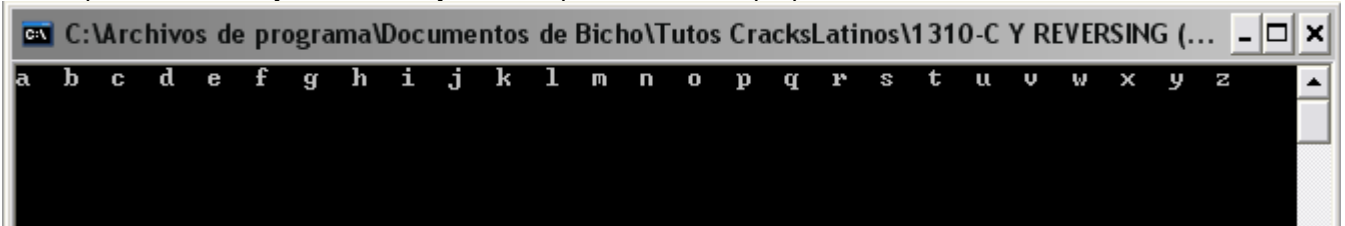


# Curso C++ y reversing de Ricardo Narvaja.

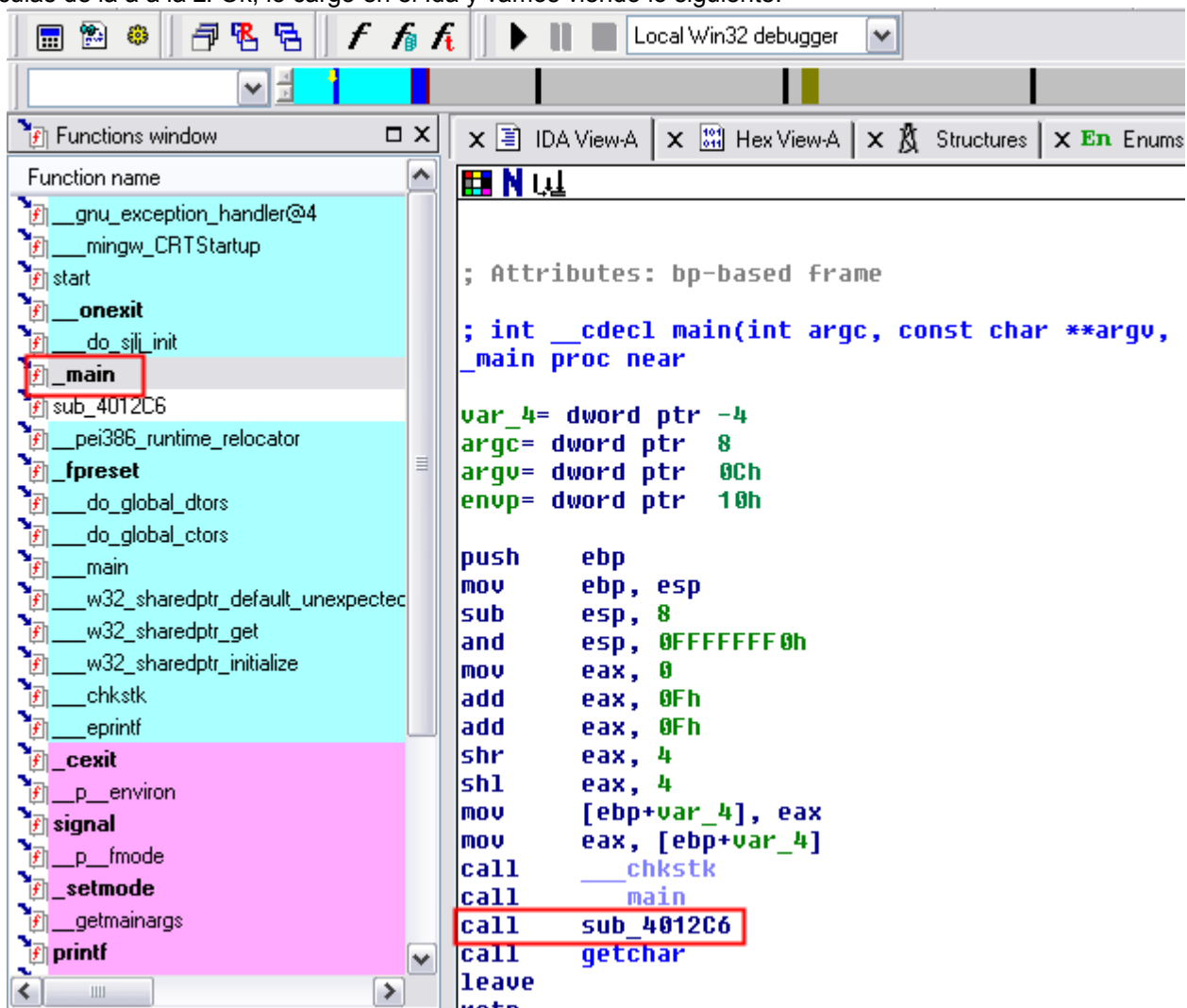
## Ejercicio 6

Saludos gente. Estamos con el curso de C++ que está escribiendo nuestro amigo Ricardo Narvaja y que está enviando a la lista por entregas. Junto con algunas entregas, nos pone unos ejercicios para ver el nivel que vamos tomando. En este tutorial les voy a intentar explicar de la mejor forma posible, cómo solucionar éste ejercicio con Ida, pues es la herramienta de depurado que se está utilizando en el curso. Después de usar Ida y comprender cómo se comporta el código, podemos escribir nosotros mismos el código en el IDE Dev-C++ hasta lograr un ejecutable que funcione exactamente igual al ejercicio propuesto por Ricardo.

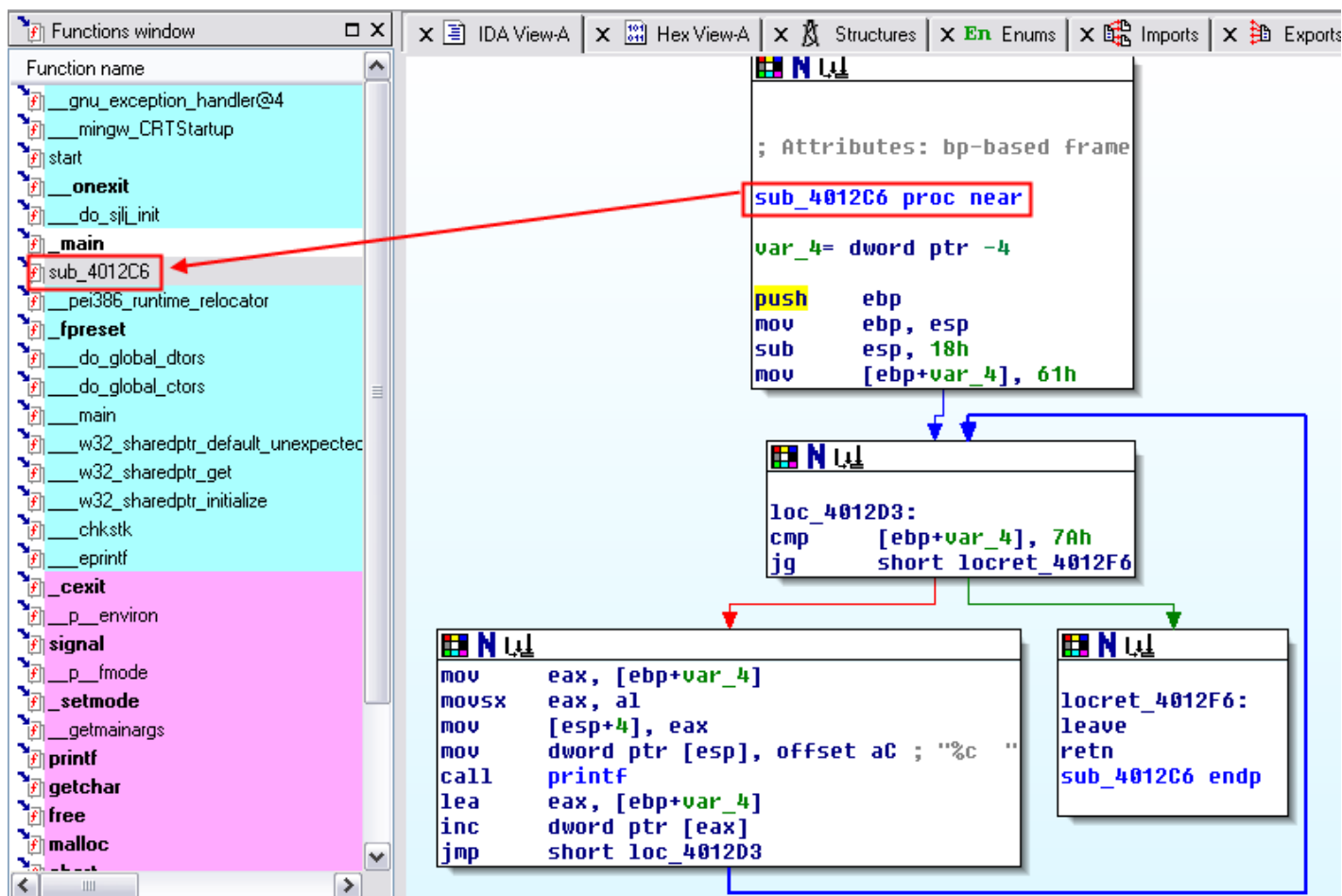
Bien, nos ponemos a ello y abrimos el ejercicio 6 que Ricardo nos propone:



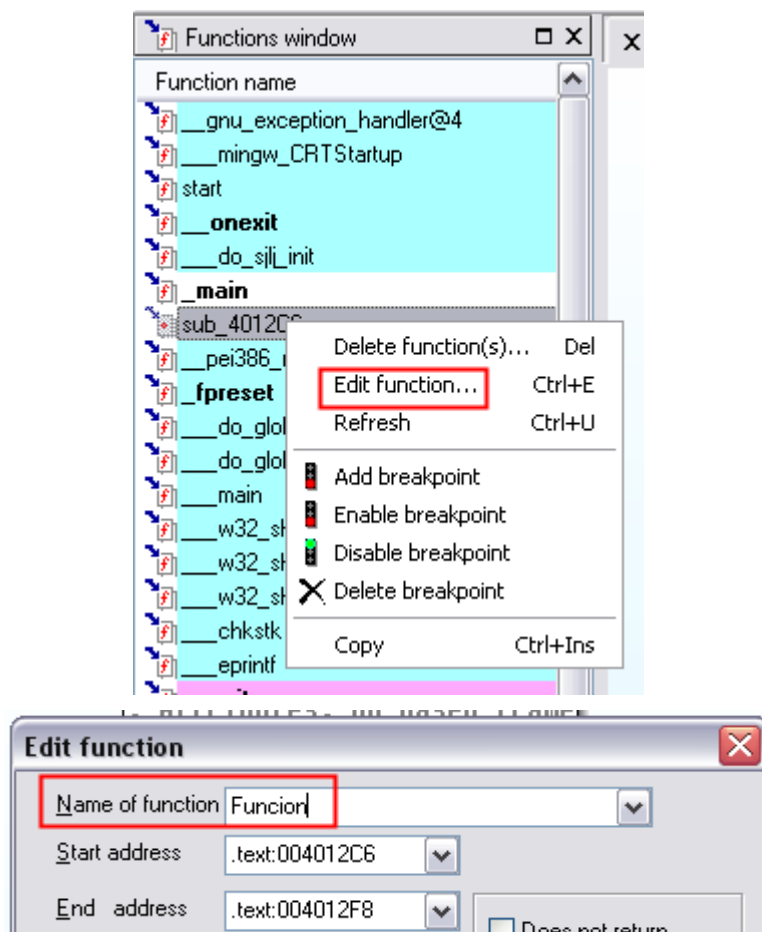
Éste pequeño programita, cuando lo ejecutamos, nos imprime en una ventana de comandos el abecedario en minúsculas de la a a la z. Ok, lo cargo en el Ida y vamos viendo lo siguiente:



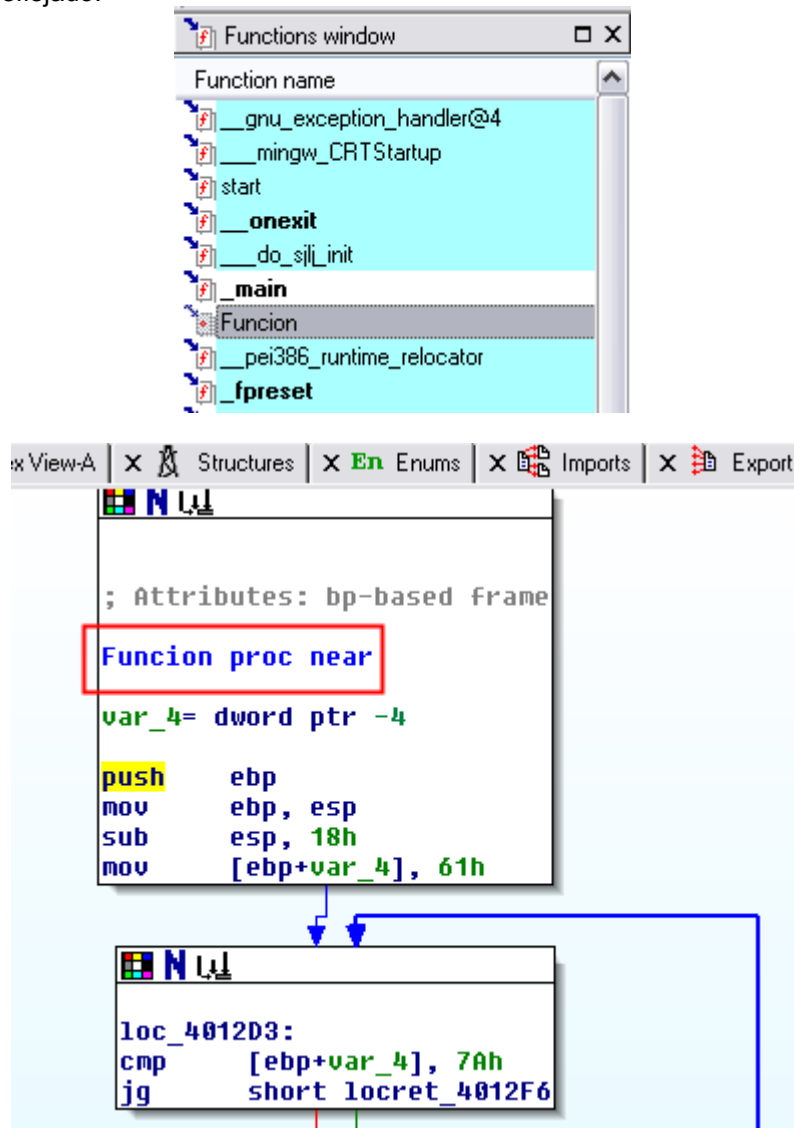
Una vez cargado en el Ida, estamos en el Main, punto de entrada o comienzo del programa. Como no suelo utilizar a menudo el Ida, no puedo detallarles de forma minuciosa todo. Para ello, deberían leer por ejemplo, el trabajo de Bigundilla. Bueno, se ve que en el código, se hace una llamada a una función. La sub\_4012C6. Vale, me voy a la ventana de funciones, le hago doble click y miro a ver de qué se trata:



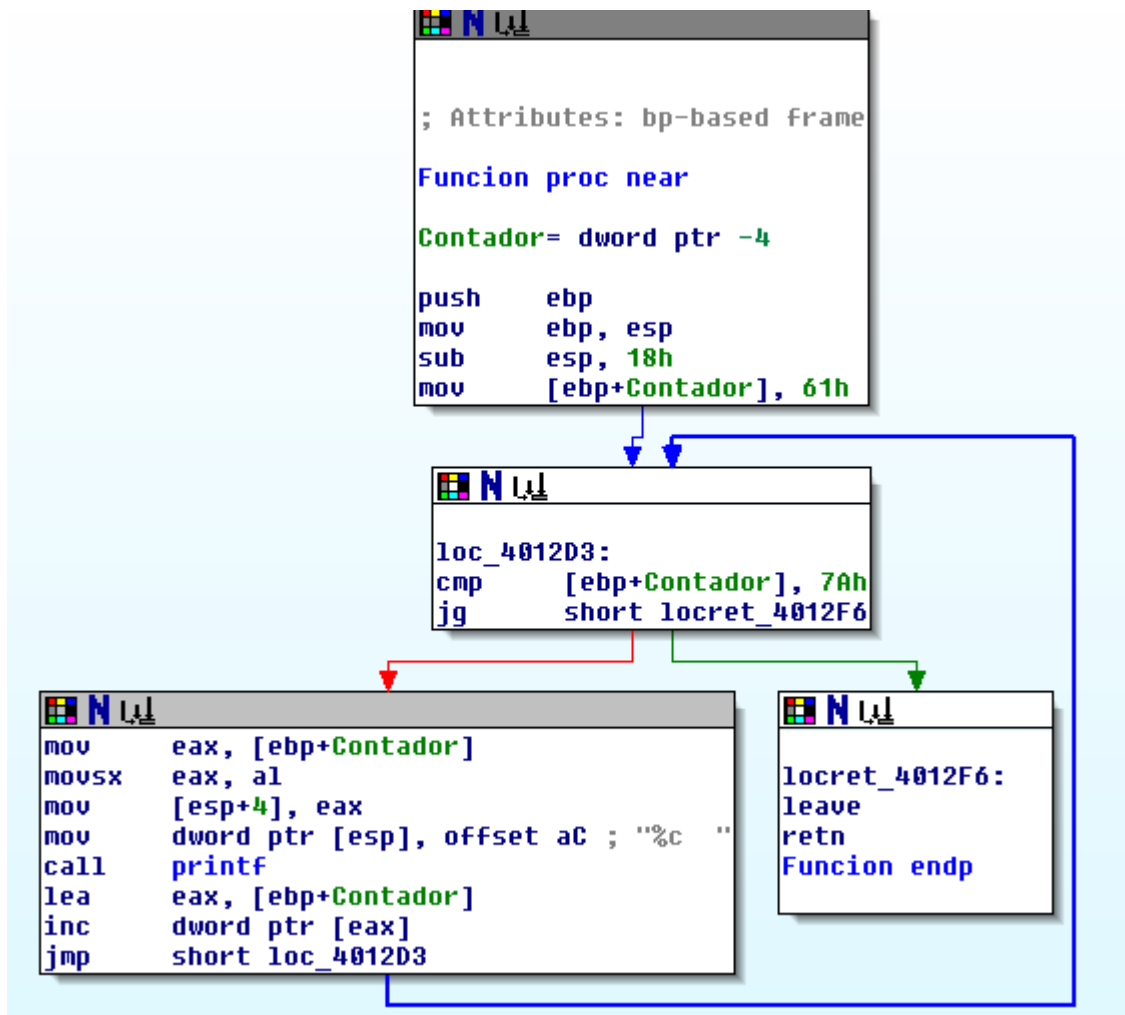
Como el programita es pequeño, no tiene nada de complicación. A parte, solo con ver lo que saca por pantalla, podemos saber qué hace el código para poner el abecedario desde la a a la z. Tal como va contándonos Ricardo en el cursillo, podemos customizar el código que vemos y hacer que nos resulte más reconocible lo que vemos. Por ejemplo si me coloco en la ventana funciones y señalo la sub\_4012C6, con el botón derecho de la ratón le podemos cambiar el nombre:



Acepto y ya me aparece reflejado:



Bien, el funcionamiento del código de este ejercicio es más que intuitivo: un contador que empieza en el hexadecimal 61, que correspondería al ASCII la letra "a" y que termina en el hexadecimal 7A que correspondería a la letra "z". Un bucle que utiliza ese contador, va colocando en pantalla las letras correspondientes a cada hexadecimal. Voy a tunear el desensamblado de Ida para que se vea claramente lo que digo. El contador, que en la primera vuelta valdrá 61h, se va depositando en la variable `var_4`, así que en Ida, cambio la `var_4` y la renombro a "Contador":



En esta imagen última, en el primer bloque de arriba, se ve cómo se mueve a la variable (espacio de memoria) Contador, el hexadecimal 61. Lo siguiente, se compara si el contenido del contador, es el hexadecimal 7A. Mientras el contenido de la variable contador no sea mayor que 7A, se continúa por la flechita roja. Se toma el hexadecimal 61 y se utiliza el formateo de carácter más un espacio, osea "%c ", para utilizarlo con el printf y sacarlo por la ventana de comandos. Obviamente, el carácter del hexadecimal 61 es la "a" minúscula. Luego, se incrementa el contador en uno y se salta a la loc\_4012D3. Se vuelve a comparar el valor del contador, que en esta ocasión sería 62, con 7A. Como todavía NO es mayor que 7A, se vuelve a ejecutar el bucle y printf utiliza el carácter del hexadecimal 62 ("b") más un espacio y lo saca en la misma ventana de comandos, seguido del anterior "a ". En el formateo, al no existir el operador "\n", no hay salto de línea y por eso se continúa en la misma línea de la ventana de comandos. Y básicamente esto es lo que hace el código. Es todo muy sencillo para que se empiece poco a poco entendiéndolo todo. Sin complicaciones todavía. Como ya sabemos cómo funciona el código y qué hace, nos ponemos manos a la obra y escribimos el código necesario. Yo, en el Ide Dev-C++ lo planteo así de fácil:

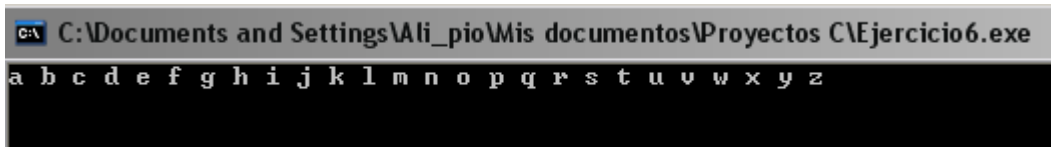
```

Ejercicio6.cpp

#include <stdio.h>
main()
{
    int LimiteInferior;
    int LimiteSuperior;
    int Contador;
    LimiteInferior=97; /*Yo prefiero trabajar con decimales :-D */
    LimiteSuperior=122;
    for(Contador=LimiteInferior; Contador<=LimiteSuperior; Contador=Contador+1)
    printf("%c ",Contador);
    getchar();
}

```

Como ven, en el comentario, yo tomo los valores en decimal. No se, cuestión de costumbre. Bien, compiló y ejecuto este código y obtengo el mismo resultado que propuso Ricardo:



Y bueno, ejercicio terminado con exito.

Un saludo, ánimos a quienes estén con el cursillo y gracias a Ricardo por dedicarle tiempo a elaborar el curso.  
Hasta la proxima!!

sequeyo  
sequeyo