

75.16 Lenguajes de Programación

PL/0

★★★ INFINITE ★★★

Dario Seminara

0. ¿De donde sale el nombre? ¿Que es eso de “INFINITE”?

Justo cuando empecé el desarrollo de PL/0 Infinite, estaba jugando un juego que me gusta bastante que se llama “Bioshock: Infinite” (el tercero de la saga), y de ahí el *infinite* como nombre característico.

Para justificar esto, se agregó la palabra reservada “INFINITE” que hace un bucle infinito, nada más:

INFINITE DO
<statement>

Ejemplo:

```
INFINITE DO writeln('HELLO').
```

1. Descripcion del compilador

PL/0 Infinite esta desarrollado 100% como una aplicacion en el browser desarrollada en javascript que **no tiene componente de servidor**, esto implica un cambio funcional significativo en comparacion a un compilador que sea por linea de comandos:

- El listado codigo fuente de entrada no puede ser un archivo del filesystem de la computadora (innaccesible para una pagina web),
- El ejecutable de salida, no se escribe directamente al filesystem, sino que es “descargado”
- Se provee una UI web con un editor de textos y comandos para ejecutar/compilar a EXE

Pero al mismo tiempo, esta eleccion de tecnologias tiene sus ventajas:

- Dependencias simplificadas (solo se requiere un browser compatible, como Google Chrome)
- No hay proceso de instalacion, salvo por un web browser si es necesario
- Portabilidad entre distintos OS mientras se mantenga el mismo browser (y portabilidad entre browsers con adaptaciones/cambios minimos)
- Facil de debuggear

Division en modulos

Para simplificar el desarrollo, se dividio el trabajo en modulos con entradas y salidas bien definidas de forma que se pueda usar Test Driven Development. Con algunas excepciones, siendo la mas importante el **compilador**, que produce una salida (el ejecutable) que es muy dificil de probar automaticamente.

	Input	Output	TDD
Scanner	String de CODIGO FUENTE	Secuencia de tokens	SI
Parser	Secuencia de tokens	Arbol de sintaxis	SI
Analizador Semantico	Arbol de sintaxis	Arbol de sintaxis digerido	SI
Compilador	Arbol de sintaxis digerido	Instrucciones de ensamblador	NO
Ensamblador	Instrucciones de ensamblador	EJECUTABLE	SI (Para funcionalidades clave)

Scanner

https://github.com/tario/pl0_infinite/blob/gh-pages/src/pl0_infinite.js#L69

Lee la cadena de texto conteniendo el código fuente en lenguaje PL/0, y obtiene los tokens que se representan ahí, así por ej

WRITELN('hello world')

Es identificado como **cuatro** tokens:

1. **WRITELN**
2. **(**
3. **'hello world'**
4. **)**

Parser

https://github.com/tario/pl0_infinite/blob/gh-pages/src/pl0_infinite.js#L121

A partir de los tokens que lee el scanner, el parser reconoce estructuras de sintaxis propias del lenguaje PL/0, así en el ejemplo anterior del writeln, el parser es capaz de reconocer que se trata de una **sentencia** del tipo writeln con un **string** "hello world". Claramente debe reconocer estructuras más complejas como whiles, procedures, etc...

La salida del parser es un **árbol de sintaxis** enviado incrementalmente en forma de mensajes a otro objeto de salida (el analizador sintáctico)

Analizador Semántico

https://github.com/tario/pl0_infinite/blob/gh-pages/src/pl0_infinite.js#L385

A partir del árbol sintáctico obtenido por el parser, el analizador semántico "digiere", es decir, elimina redundancias y simplifica todo lo posible las representaciones según la semántica del lenguaje:

- Reemplaza el uso de las constantes por su valor declarado
- Reemplaza el uso de las variables con un offset numérico, eliminando además las ambigüedades (ej: dos variables que se llaman igual en distinto procedure, obtendrán distinto offset)
- Idem para los procedures
- Substituye el write o writeln con múltiples argumentos, por múltiples write o writeln

Compilador

https://github.com/tario/pl0_infinite/blob/gh-pages/src/pl0_infinite/compiler.js

A partir del arbol sintatico digerido, traduce esa representacion a assembler x86 intel de 32 bit, ademas se encarga de generar y configurar el header para un ejecutable de Win32 (**PE**)

Las instrucciones de ensamblador las envia al modulo "Ensamblador" para que este vaya generando los opcodes/direcciones/etc.. correspondientes

Ensamblador

https://github.com/tario/pl0_infinite/blob/gh-pages/src/pl0_infinite/asm.js

Recibe las instrucciones en ensamblador del compilador, y genera las instrucciones en codigo maquina corespondiente, ademas se encarga de calcular los offsets de saltos relativos

Particularidades

Todo esta escrito en Javascript

Esto ya se menciono, no hay ningun componente que se ejecute fuera del entorno de ejecucion del browser (salvo obviamente los ejecutables generados)

Los ejecutables se generan con blobs

Blob es un grupo de API del browser que permite generar archivos (incluyendo binarios) en la memoria del propio browser, y ademas permite generar links de descarga a ese archivo virtual mediante [createObjectURL](#) (por lo que no hay necesidad de depender de un servidor desde donde descargar el producto de la compilacion)

Mas informacion: <https://developer.mozilla.org/es/docs/Web/API/Blob>

La evaluación de expresiones se hace primariamente entre registros

Inicialmente se planteo utilizar el **stack** para almacenar temporalmente el resultado de los componentes de una expresion a cualquier nivel, con lo que una expresion de suma **X+5**, se compilaria asi:

```
mov eax, [edi+0] # leer X
push eax        # enviarlo al stack
push 5          # enviar la constante '5' al stack
pop eax         # retirar dos valores del stack
pop ebx
add eax, ebx    # sumar los valores obtenidos
push eax       # enviar el resultado al stack
```

Pero en lugar de eso se opto por simplemente usar los registros para que sea mas “*intel style*”, por ejemplo una suma de una variable con un numero (**X+5**) se compilaria a algo como esto:

```
mov eax, [edi+0] # leer X
mov ebx, eax
mov eax, 5       # colocar 5 en eax
add ebx, eax
mov eax, ebx    # sumar
```

Este metodo encuentra su limite, cuando se tienen que evaluar expresiones anidadas que tambien hacen uso de los mismos registros, es ahi cuando se recurre al stack para salvar esos registros, asi por ej $X+5*Y$, se compilaria asi:

```

mov eax, [edi+0] # leer X
mov ebx, eax

push ebx        # salvar ebx porque sera usado para
                # calcular 5*Y

mov eax, 5      # colocar 5 en eax
mov ebx, eax    # pasarlo a ebx
mov eax, [edi+4] # leer Y
imul eax, ebx   # multiplicar ebx por el valor devuelto
mov eax, ebx

pop ebx         # restaurar ebx que se salvo previamente

add ebx, eax    # sumar el valor devuelto por 5*Y

```

Claramente esto se puede optimizar mejor aprovechando todos los registros, pero en primera instancia ahorra tambien la necesidad de patchear las optimizaciones relativas al uso del stack (por ej **push eax; pop eax**)

https://github.com/tario/pl0_infinite/blob/gh-pages/src/pl0_infinite/compiler.js#L138

No se implemento un mecanismo de recuperacion ante errores

El proceso de compilacion solo informa del primer error que encuentra, pero a cambio se ejecuta tan rapido que puede compilar incluso ante cada minimo cambio (aunque sea solo un caracter mas) en el codigo fuente que se haga en el editor, mostrando de manera instantanea los errores sintacticos y semanticos que se detecta a medida que se escribe/modifica el codigo fuente del programa

2. Programas correctos

00

```
var X, Y;

procedure INICIAR;
  const Y = 2;
  procedure ASIGNAR;
    X := Y;
  call ASIGNAR;

begin
  write ('NUM='); readln (Y);
  call INICIAR;
  writeln ('NUM*2=', Y*X)
end.
```



```

000003d0    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
000003e0    60 89 c6 30 c0 02 06 74    08 46 e8 e1 fe ff ff eb    `..0...t.F.....
000003f0    f2 61 90 c3 00 00 00 00    00 00 00 00 00 00 00 00    .a.....
00000400    04 30 e8 c9 fe ff ff c3    00 00 00 00 00 00 00 00    .0.....
00000410    b0 0d e8 b9 fe ff ff b0    0a e8 b2 fe ff ff c3 00    .....
00000420    3d 00 00 00 80 75 4e b0    2d e8 a2 fe ff ff b0 02    =....uN.-.....
00000430    e8 cb ff ff ff b0 01 e8    c4 ff ff ff b0 04 e8 bd    .....
00000440    ff ff ff b0 07 e8 b6 ff    ff ff b0 04 e8 af ff ff    .....
00000450    ff b0 08 e8 a8 ff ff ff    b0 03 e8 a1 ff ff ff b0    .....
00000460    06 e8 9a ff ff ff b0 04    e8 93 ff ff ff b0 08 e8    .....
00000470    8c ff ff ff c3 3d 00 00    00 00 7d 0b 50 b0 2d e8    ....=....}.P.-.
00000480    4c fe ff ff 58 f7 d8 3d    0a 00 00 00 0f 8c ef 00    L...X..=.....
00000490    00 00 3d 64 00 00 00 0f    8c d1 00 00 00 3d e8 03    ..=d.....=..
000004a0    00 00 0f 8c b3 00 00 00    3d 10 27 00 00 0f 8c 95    .....='.....
000004b0    00 00 00 3d a0 86 01 00    7c 7b 3d 40 42 0f 00 7c    ...=....|{@B..|
000004c0    61 3d 80 96 98 00 7c 47    3d 00 e1 f5 05 7c 2d 3d    a=....|G=....|-
000004d0    00 ca 9a 3b 7c 13 ba 00    00 00 00 bb 00 ca 9a 3b    ...;|.....;
000004e0    f7 fb 52 e8 18 ff ff ff    58 ba 00 00 00 00 bb 00    ..R....X.....
000004f0    e1 f5 05 f7 fb 52 e8 05    ff ff ff 58 ba 00 00 00    ....R....X....
00000500    00 bb 80 96 98 00 f7 fb    52 e8 f2 fe ff ff 58 ba    .....R....X.
00000510    00 00 00 00 bb 40 42 0f    00 f7 fb 52 e8 df fe ff    .....@B....R....
00000520    ff 58 ba 00 00 00 00 bb    a0 86 01 00 f7 fb 52 e8    .X.....R.
00000530    cc fe ff ff 58 ba 00 00    00 00 bb 10 27 00 00 f7    ....X.....'...
00000540    fb 52 e8 b9 fe ff ff 58    ba 00 00 00 00 bb e8 03    .R....X.....
00000550    00 00 f7 fb 52 e8 a6 fe    ff ff 58 ba 00 00 00 00    ....R....X.....
00000560    bb 64 00 00 00 f7 fb 52    e8 93 fe ff ff 58 ba 00    .d....R....X..
00000570    00 00 00 bb 0a 00 00 00    f7 fb 52 e8 80 fe ff ff    .....R.....
00000580    58 e8 7a fe ff ff c3 00    ff 15 00 10 40 00 00 00    X.z.....@...
00000590    b9 00 00 00 00 b3 03 51    53 e8 a2 fd ff ff 5b 59    .....QS.....[Y
000005a0    3c 0d 0f 84 34 01 00 00    3c 08 0f 84 94 00 00 00    <...4...<.....
000005b0    3c 2d 0f 84 09 01 00 00    3c 30 7c db 3c 39 7f d7    <-.....<0|.<9..
000005c0    2c 30 80 fb 00 74 d0 80    fb 02 75 0c 81 f9 00 00    ,0...t....u.....
000005d0    00 00 75 04 3c 00 74 bf    80 fb 03 75 0a 3c 00 75    ..u.<.t....u.<.u
000005e0    04 b3 00 eb 02 b3 01 81    f9 cc cc cc 0c 7f a8 81    .....
000005f0    f9 34 33 33 f3 7c a0 88    c7 b8 0a 00 00 00 f7 e9    .433.|.....
00000600    3d 08 00 00 80 74 11 3d    f8 ff ff 7f 75 13 80 ff    =....t.=....u...
00000610    07 7e 0e e9 7f ff ff ff    80 ff 08 0f 8f 76 ff ff    .~.....v..
00000620    ff b9 00 00 00 00 88 f9    80 fb 02 74 04 01 c1 eb    .....t....
00000630    03 29 c8 91 88 f8 51 53    e8 c3 fd ff ff 5b 59 e9    .)....QS.....[Y.
00000640    53 ff ff ff 80 fb 03 0f    84 4a ff ff ff 51 53 b0    S.....J...QS.
00000650    08 e8 7a fc ff ff b0 20    e8 73 fc ff ff b0 08 e8    ..z....s.....
00000660    6c fc ff ff 5b 59 80 fb    00 75 07 b3 03 e9 25 ff    l...[Y...u....%.
00000670    ff ff 80 fb 02 75 0f 81    f9 00 00 00 00 75 07 b3    ....u.....u..
00000680    03 e9 11 ff ff ff 89 c8    b9 0a 00 00 00 ba 00 00    .....
00000690    00 00 3d 00 00 00 00 7d    08 f7 d8 f7 f9 f7 d8 eb    ..=....}.....
000006a0    02 f7 f9 89 c1 81 f9 00    00 00 00 0f 85 e6 fe ff    .....
000006b0    ff 80 fb 02 0f 84 dd fe    ff ff b3 03 e9 d6 fe ff    .....
000006c0    ff 80 fb 03 0f 85 cd fe    ff ff b0 2d 51 53 e8 fd    .....-QS..
000006d0    fb ff ff 5b 59 b3 02 e9    bb fe ff ff 80 fb 03 0f    ...[Y.....
000006e0    84 b2 fe ff ff 80 fb 02    75 0c 81 f9 00 00 00 00    .....u.....
000006f0    0f 84 a1 fe ff ff 51 e8    14 fd ff ff 59 89 c8 c3    .....Q....Y...
00000700    bf 67 15 40 00 e9 17 00    00 00 e9 0c 00 00 00 b8    .g.@.....
00000710    02 00 00 00 89 87 00 00    00 00 c3 e8 ef ff ff ff    .....
00000720    c3 b8 6f 15 40 00 e8 b5    fc ff ff e8 60 fe ff ff    ..o.@.....`...
00000730    89 87 04 00 00 00 e8 e0    ff ff ff b8 74 15 40 00    .....t.@.
00000740    e8 9b fc ff ff 8b 87 04    00 00 00 89 c3 8b 87 00    .....
00000750    00 00 00 0f af d8 89 d8    e8 c3 fc ff ff e8 ae fc    .....
00000760    ff ff e9 21 fe ff ff 00    00 00 00 00 00 00 00 4e    ...!.....N
00000770    55 4d 3d 00 4e 55 4d 2a    32 3d 00 00 00 00 00 00    UM=.NUM*2=.....
00000780    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
00000790    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
000007a0    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....

```

000007b0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000007c0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000007d0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000007e0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000007f0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

desensamblado

```

:00401500 BF67154000      mov edi, 00401567
:00401505 E917000000      jmp 00401521
:0040150A E90C000000      jmp 0040151B

```

* Referenced by a CALL at Address:

|:0040151B

|

```

:0040150F B802000000      mov eax, 00000002
:00401514 898700000000      mov dword ptr [edi+00000000], eax
:0040151A C3              ret

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:0040150A(U)

|

```

:0040151B E8FFFFFFF      call 0040150F
:00401520 C3              ret

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401505(U)

|

```

:00401521 B86F154000      mov eax, 0040156F
:00401526 E8B5FCFFFF      call 004011E0
:0040152B E860FEFFFF      call 00401390
:00401530 898704000000      mov dword ptr [edi+00000004], eax
:00401536 E8E0FFFFFF      call 0040151B
:0040153B B874154000      mov eax, 00401574
:00401540 E89BFCFFFF      call 004011E0
:00401545 8B8704000000      mov eax, dword ptr [edi+00000004]
:0040154B 89C3              mov ebx, eax
:0040154D 8B8700000000      mov eax, dword ptr [edi+00000000]
:00401553 0FAFD8           imul ebx, eax
:00401556 89D8              mov eax, ebx
:00401558 E8C3FCFFFF      call 00401220
:0040155D E8AEFCFFFF      call 00401210
:00401562 E921FEFFFF      jmp 00401388
:00401567 0000000000000000  BYTE 8 DUP(0)

```

```

:0040156F 4E              dec esi
:00401570 55              push ebp

```

```
:00401571 4D          dec ebp
:00401572 3D004E554D  cmp eax, 4D554E00
:00401577 2A32        sub dh, byte ptr [edx]
:00401579 3D00000000  cmp eax, 00000000
:0040157E 00000000000000000000000000000000  BYTE 10 DUP(0)
```

01

```
VAR BASE, EXPO, RESU;

PROCEDURE POT;
IF EXPO > 0 THEN
  BEGIN
    RESU := RESU * BASE;
    EXPO := EXPO - 1;
    CALL POT
  END;

BEGIN
  WRITE ('BASE: '); READLN(BASE);
  WRITE ('EXPONENTE: '); READLN(EXPO);
  RESU := 1;
  CALL POT;
  IF EXPO < 0 THEN RESU := 0;
  WRITELN ('RESULTADO: ', RESU);
  WRITELN
END.
```



```

000003d0    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
000003e0    60 89 c6 30 c0 02 06 74    08 46 e8 e1 fe ff ff eb    `..0...t.F.....
000003f0    f2 61 90 c3 00 00 00 00    00 00 00 00 00 00 00 00    .a.....
00000400    04 30 e8 c9 fe ff ff c3    00 00 00 00 00 00 00 00    .0.....
00000410    b0 0d e8 b9 fe ff ff b0    0a e8 b2 fe ff ff c3 00    .....
00000420    3d 00 00 00 80 75 4e b0    2d e8 a2 fe ff ff b0 02    =....uN.-.....
00000430    e8 cb ff ff ff b0 01 e8    c4 ff ff ff b0 04 e8 bd    .....
00000440    ff ff ff b0 07 e8 b6 ff    ff ff b0 04 e8 af ff ff    .....
00000450    ff b0 08 e8 a8 ff ff ff    b0 03 e8 a1 ff ff ff b0    .....
00000460    06 e8 9a ff ff ff b0 04    e8 93 ff ff ff b0 08 e8    .....
00000470    8c ff ff ff c3 3d 00 00    00 00 7d 0b 50 b0 2d e8    .....=....}.P.-.
00000480    4c fe ff ff 58 f7 d8 3d    0a 00 00 00 0f 8c ef 00    L...X..=.....
00000490    00 00 3d 64 00 00 00 0f    8c d1 00 00 00 3d e8 03    ..=d.....=.
000004a0    00 00 0f 8c b3 00 00 00    3d 10 27 00 00 0f 8c 95    .....=.'.....
000004b0    00 00 00 3d a0 86 01 00    7c 7b 3d 40 42 0f 00 7c    ...=....|{@B..|
000004c0    61 3d 80 96 98 00 7c 47    3d 00 e1 f5 05 7c 2d 3d    a=....|G=....|-=
000004d0    00 ca 9a 3b 7c 13 ba 00    00 00 00 bb 00 ca 9a 3b    ...;|.....;
000004e0    f7 fb 52 e8 18 ff ff ff    58 ba 00 00 00 00 bb 00    ..R....X.....
000004f0    e1 f5 05 f7 fb 52 e8 05    ff ff ff 58 ba 00 00 00    .....R....X....
00000500    00 bb 80 96 98 00 f7 fb    52 e8 f2 fe ff ff 58 ba    .....R....X.
00000510    00 00 00 00 bb 40 42 0f    00 f7 fb 52 e8 df fe ff    .....@B....R....
00000520    ff 58 ba 00 00 00 00 bb    a0 86 01 00 f7 fb 52 e8    .X.....R.
00000530    cc fe ff ff 58 ba 00 00    00 00 bb 10 27 00 00 f7    ....X.....'...
00000540    fb 52 e8 b9 fe ff ff 58    ba 00 00 00 00 bb e8 03    .R....X.....
00000550    00 00 f7 fb 52 e8 a6 fe    ff ff 58 ba 00 00 00 00    ...R....X.....
00000560    bb 64 00 00 00 f7 fb 52    e8 93 fe ff ff 58 ba 00    .d.....R....X..
00000570    00 00 00 bb 0a 00 00 00    f7 fb 52 e8 80 fe ff ff    .....R.....
00000580    58 e8 7a fe ff ff c3 00    ff 15 00 10 40 00 00 00    X.z.....@...
00000590    b9 00 00 00 00 b3 03 51    53 e8 a2 fd ff ff 5b 59    .....QS.....[Y
000005a0    3c 0d 0f 84 34 01 00 00    3c 08 0f 84 94 00 00 00    <...4...<.....
000005b0    3c 2d 0f 84 09 01 00 00    3c 30 7c db 3c 39 7f d7    <-.....<0|. <9..
000005c0    2c 30 80 fb 00 74 d0 80    fb 02 75 0c 81 f9 00 00    ,0...t....u.....
000005d0    00 00 75 04 3c 00 74 bf    80 fb 03 75 0a 3c 00 75    ..u.<.t....u.<.u
000005e0    04 b3 00 eb 02 b3 01 81    f9 cc cc cc 0c 7f a8 81    .....
000005f0    f9 34 33 33 f3 7c a0 88    c7 b8 0a 00 00 00 f7 e9    .433.|.....
00000600    3d 08 00 00 80 74 11 3d    f8 ff ff 7f 75 13 80 ff    =....t.=...u...
00000610    07 7e 0e e9 7f ff ff ff    80 ff 08 0f 8f 76 ff ff    .~.....v..
00000620    ff b9 00 00 00 00 88 f9    80 fb 02 74 04 01 c1 eb    .....t....
00000630    03 29 c8 91 88 f8 51 53    e8 c3 fd ff ff 5b 59 e9    .)....QS.....[Y.
00000640    53 ff ff ff 80 fb 03 0f    84 4a ff ff ff 51 53 b0    S.....J....QS.
00000650    08 e8 7a fc ff ff b0 20    e8 73 fc ff ff b0 08 e8    .z.... .s.....
00000660    6c fc ff ff 5b 59 80 fb    00 75 07 b3 03 e9 25 ff    l...[Y...u....%.
00000670    ff ff 80 fb 02 75 0f 81    f9 00 00 00 00 75 07 b3    .....u.....u..
00000680    03 e9 11 ff ff ff 89 c8    b9 0a 00 00 00 ba 00 00    .....
00000690    00 00 3d 00 00 00 00 7d    08 f7 d8 f7 f9 f7 d8 eb    ..=....}).....
000006a0    02 f7 f9 89 c1 81 f9 00    00 00 00 0f 85 e6 fe ff    .....
000006b0    ff 80 fb 02 0f 84 dd fe    ff ff b3 03 e9 d6 fe ff    .....
000006c0    ff 80 fb 03 0f 85 cd fe    ff ff b0 2d 51 53 e8 fd    .....-QS..
000006d0    fb ff ff 5b 59 b3 02 e9    bb fe ff ff 80 fb 03 0f    ...[Y.....
000006e0    84 b2 fe ff ff 80 fb 02    75 0c 81 f9 00 00 00 00    .....u.....
000006f0    0f 84 a1 fe ff ff 51 e8    14 fd ff ff 59 89 c8 c3    .....Q.....Y...
00000700    bf dd 15 40 00 e9 4b 00    00 00 8b 87 04 00 00 00    ...@..K.....
00000710    89 c3 b8 00 00 00 00 39    c3 0f 8e 35 00 00 00 8b    .....9...5....
00000720    87 08 00 00 00 89 c3 8b    87 00 00 00 00 0f af d8    .....
00000730    89 d8 89 87 08 00 00 00    8b 87 04 00 00 00 89 c3    .....
00000740    b8 01 00 00 00 29 c3 89    d8 89 87 04 00 00 00 e8    .....).....
00000750    b6 ff ff ff c3 b8 e9 15    40 00 e8 81 fc ff ff e8    .....@.....
00000760    2c fe ff ff 89 87 00 00    00 00 b8 f0 15 40 00 e8    ,.....@..
00000770    6c fc ff ff e8 17 fe ff    ff 89 87 04 00 00 00 b8    l.....
00000780    01 00 00 00 89 87 08 00    00 00 e8 7b ff ff ff 8b    .....{....
00000790    87 04 00 00 00 89 c3 b8    00 00 00 00 39 c3 0f 8d    .....9...
000007a0    0b 00 00 00 b8 00 00 00    00 89 87 08 00 00 00 b8    .....

```

000007b0	fc 15 40 00 e8 27 fc ff	ff 8b 87 08 00 00 00 e8	..@...'.....
000007c0	5c fc ff ff e8 47 fc ff	ff b8 08 16 40 00 e8 0d	\....G.....@...
000007d0	fc ff ff e8 38 fc ff ff	e9 ab fd ff ff 00 00 008.....
000007e0	00 00 00 00 00 00 00 00	00 42 41 53 45 3a 20 00BASE: .
000007f0	45 58 50 4f 4e 45 4e 54	45 3a 20 00 52 45 53 55	EXPONENTE: .RESU
00000800	4c 54 41 44 4f 3a 20 00	00 00 00 00 00 00 00 00	LTADO:
00000810	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000820	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000830	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000840	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000850	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000860	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000870	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000880	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000890	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008a0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008b0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008c0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008d0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008e0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008f0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

desensamblado

```

:00401500 BFDD154000      mov edi, 004015DD
:00401505 E94B000000      jmp 00401555
:0040150A 8B8704000000      mov eax, dword ptr [edi+00000004]
:00401510 89C3              mov ebx, eax
:00401512 B800000000      mov eax, 00000000
:00401517 39C3              cmp ebx, eax
:00401519 0F8E35000000      jle 00401554
:0040151F 8B8708000000      mov eax, dword ptr [edi+00000008]
:00401525 89C3              mov ebx, eax
:00401527 8B8700000000      mov eax, dword ptr [edi+00000000]
:0040152D 0FAFD8           imul ebx, eax
:00401530 89D8              mov eax, ebx
:00401532 898708000000      mov dword ptr [edi+00000008], eax
:00401538 8B8704000000      mov eax, dword ptr [edi+00000004]
:0040153E 89C3              mov ebx, eax
:00401540 B801000000      mov eax, 00000001
:00401545 29C3              sub ebx, eax
:00401547 89D8              mov eax, ebx
:00401549 898704000000      mov dword ptr [edi+00000004], eax
:0040154F E8B6FFFFFF       call 0040150A

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401519(C)

```

|
:00401554 C3              ret

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401505(U)

```

|
:00401555 B8E9154000      mov eax, 004015E9
:0040155A E881FCFFFF       call 004011E0
:0040155F E82CFEFFFF       call 00401390
:00401564 898700000000      mov dword ptr [edi+00000000], eax
:0040156A B8F0154000      mov eax, 004015F0
:0040156F E86CFCFFFF       call 004011E0
:00401574 E817FEFFFF       call 00401390
:00401579 898704000000      mov dword ptr [edi+00000004], eax
:0040157F B801000000      mov eax, 00000001
:00401584 898708000000      mov dword ptr [edi+00000008], eax
:0040158A E87BFFFFFF       call 0040150A
:0040158F 8B8704000000      mov eax, dword ptr [edi+00000004]
:00401595 89C3              mov ebx, eax
:00401597 B800000000      mov eax, 00000000
:0040159C 39C3              cmp ebx, eax

```

```

:0040159E 0F8D0B000000    jnl 004015AF
:004015A4 B800000000    mov eax, 00000000
:004015A9 898708000000    mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040159E(C)

```

|
:004015AF B8FC154000    mov eax, 004015FC
:004015B4 E827FCFFFF    call 004011E0
:004015B9 8B8708000000    mov eax, dword ptr [edi+00000008]
:004015BF E85CFCFFFF    call 00401220
:004015C4 E847FCFFFF    call 00401210
:004015C9 B808164000    mov eax, 00401608
:004015CE E80DFCFFFF    call 004011E0
:004015D3 E838FCFFFF    call 00401210
:004015D8 E9ABFDFFFF    jmp 00401388
:004015DD 00000000000000000000    BYTE 10 DUP(0)
:004015E7 0000    BYTE 2 DUP(0)

```

```

:004015E9 42    inc edx
:004015EA 41    inc ecx
:004015EB 53    push ebx
:004015EC 45    inc ebp
:004015ED 3A20    cmp ah, byte ptr [eax]
:004015EF 004558    add byte ptr [ebp+58], al
:004015F2 50    push eax
:004015F3 4F    dec edi
:004015F4 4E    dec esi
:004015F5 45    inc ebp
:004015F6 4E    dec esi
:004015F7 54    push esp
:004015F8 45    inc ebp
:004015F9 3A20    cmp ah, byte ptr [eax]
:004015FB 005245    add byte ptr [edx+45], dl
:004015FE 53    push ebx
:004015FF 55    push ebp
:00401600 4C    dec esp
:00401601 54    push esp
:00401602 41    inc ecx
:00401603 44    inc esp
:00401604 4F    dec edi
:00401605 3A20    cmp ah, byte ptr [eax]
:00401607 00000000000000000000    BYTE 10 DUP(0)

```


02

```

var X, Y, Z;

procedure MULTIPLICAR;
var A, B;
begin
    A := X;
    B := Y;
    Z := 0;
    if X < 0 then A := -A;
    if Y < 0 then B := -B;
    while B > 0 do
        begin
            if odd B then Z:= Z + A;
            A := A * 2;
            B := B / 2
        end;
    if X < 0 then Z:= -Z;
    if Y < 0 then Z:= -Z
end;

Begin
    write ('X: '); readLn (X);
    write ('Y: '); readLn (Y);
    call MULTIPLICAR;
    writeLn ('X*Y=', Z);
end.

```

Offset	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	
00000000	4d	5a	60	01	01	00	00	00	04	00	00	00	ff	ff	00	00	MZ`.....
00000010	60	01	00	00	00	00	00	00	40	00	00	00	00	00	00	00	`.....@.....
00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000030	00	00	00	00	00	00	00	00	00	00	00	a0	00	00	00	00
00000040	0e	1f	ba	0e	00	b4	09	cd	21	b8	01	4c	cd	21	54	68!...L!Th
00000050	69	73	20	70	72	6f	67	72	61	6d	20	69	73	20	61	20	is program is a
00000060	57	69	6e	33	32	20	63	6f	6e	73	6f	6c	65	20	61	70	Win32 console ap
00000070	70	6c	69	63	61	74	69	6f	6e	2e	20	49	74	20	63	61	plication. It ca
00000080	6e	6e	6f	74	20	62	65	20	72	75	6e	20	75	6e	64	65	nnot be run unde
00000090	72	20	4d	53	2d	44	4f	53	2e	0d	0a	24	00	00	00	00	r MS-DOS...\$....
000000a0	50	45	00	00	4c	01	01	00	00	00	53	4c	00	00	00	00	PE..L....SL....
000000b0	00	00	00	00	e0	00	02	01	0b	01	01	00	00	06	00	00
000000c0	00	00	00	00	00	00	00	00	00	15	00	00	00	10	00	00
000000d0	00	20	00	00	00	00	40	00	00	10	00	00	00	02	00	00@.....
000000e0	04	00	00	00	00	00	00	00	04	00	00	00	00	00	00	00
000000f0	00	17	00	00	00	02	00	00	00	00	00	00	03	00	00	00
00000100	00	00	10	00	00	10	00	00	00	00	10	00	00	10	00	00
00000110	00	00	00	00	10	00	00	00	00	00	00	00	00	00	00	00
00000120	1c	10	00	00	28	00	00	00	00	00	00	00	00	00	00	00(.
00000130	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000170	00	00	00	00	00	00	00	00	00	10	00	00	1c	00	00	00
00000180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000190	00	00	00	00	00	00	00	00	2e	74	65	78	74	00	00	00text...
000001a0	00	07	00	00	00	10	00	00	00	07	00	00	00	02	00	00
000001b0	00	00	00	00	00	00	00	00	00	00	00	00	20	00	00	e0
000001c0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000001d0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000001e0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000001f0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000200	6e	10	00	00	7c	10	00	00	8c	10	00	00	98	10	00	00	n...
00000210	a4	10	00	00	b6	10	00	00	00	00	00	00	52	10	00	00R...
00000220	00	00	00	00	00	00	00	00	44	10	00	00	00	10	00	00D.....
00000230	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000240	00	00	00	00	4b	45	52	4e	45	4c	33	32	2e	64	6c	6c	...KERNEL32.dll
00000250	00	00	6e	10	00	00	7c	10	00	00	8c	10	00	00	98	10	..n...
00000260	00	00	a4	10	00	00	b6	10	00	00	00	00	00	00	00	00
00000270	45	78	69	74	50	72	6f	63	65	73	73	00	00	00	47	65	ExitProcess...Ge
00000280	74	53	74	64	48	61	6e	64	6c	65	00	00	00	00	52	65	tStdHandle....Re
00000290	61	64	46	69	6c	65	00	00	00	00	57	72	69	74	65	46	adFile....WriteF
000002a0	69	6c	65	00	00	00	47	65	74	43	6f	6e	73	6f	6c	65	ile...GetConsole
000002b0	4d	6f	64	65	00	00	00	00	53	65	74	43	6f	6e	73	6f	Mode....SetConso
000002c0	6c	65	4d	6f	64	65	00	00	00	00	00	00	00	00	00	00	leMode.....
000002d0	50	a2	1c	11	40	00	31	c0	03	05	2c	11	40	00	75	0d	P...@.1...,.@.u.
000002e0	6a	f5	ff	15	04	10	40	00	a3	2c	11	40	00	6a	00	68	j....@...@.j.h
000002f0	30	11	40	00	6a	01	68	1c	11	40	00	50	ff	15	0c	10	0.@.j.h..@.P....
00000300	40	00	09	c0	75	08	6a	00	ff	15	00	10	40	00	81	3d	@...u.j....@.=
00000310	30	11	40	00	01	00	00	00	75	ec	58	c3	00	57	72	69	0.@.....u.X..Wri
00000320	74	65	20	65	72	72	6f	72	00	00	00	00	00	00	00	00	te error.....
00000330	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000340	60	31	c0	03	05	cc	11	40	00	75	37	6a	f6	ff	15	04	`1....@.u7j....
00000350	10	40	00	a3	cc	11	40	00	68	d0	11	40	00	50	ff	15	.@....@.h..@.P..
00000360	10	10	40	00	80	25	d0	11	40	00	f9	ff	35	d0	11	40	..@..%...@.5..@
00000370	00	ff	35	cc	11	40	00	ff	15	14	10	40	00	a1	cc	11	..5..@.....@....
00000380	40	00	6a	00	68	d4	11	40	00	6a	01	68	be	11	40	00	@.j.h..@.j.h..@.
00000390	50	ff	15	08	10	40	00	09	c0	61	90	75	08	6a	00	ff	P....@...a.u.j..
000003a0	15	00	10	40	00	0f	b6	05	be	11	40	00	81	3d	d4	11	...@.....@.=..


```

000003b0 40 00 01 00 00 00 74 05 b8 ff ff ff ff c3 00 52 @.....t.....R
000003c0 65 61 64 20 65 72 72 6f 72 00 00 00 00 00 00 00 ead error.....
000003d0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000003e0 60 89 c6 30 c0 02 06 74 08 46 e8 e1 fe ff ff eb `..0...t.F.....
000003f0 f2 61 90 c3 00 00 00 00 00 00 00 00 00 00 00 .a.....
00000400 04 30 e8 c9 fe ff ff c3 00 00 00 00 00 00 00 00 .0.....
00000410 b0 0d e8 b9 fe ff ff b0 0a e8 b2 fe ff ff c3 00 .....
00000420 3d 00 00 00 80 75 4e b0 2d e8 a2 fe ff ff b0 02 =....uN.-.....
00000430 e8 cb ff ff ff b0 01 e8 c4 ff ff ff b0 04 e8 bd .....
00000440 ff ff ff b0 07 e8 b6 ff ff ff b0 04 e8 af ff ff .....
00000450 ff b0 08 e8 a8 ff ff ff b0 03 e8 a1 ff ff ff b0 .....
00000460 06 e8 9a ff ff ff b0 04 e8 93 ff ff ff b0 08 e8 .....
00000470 8c ff ff ff c3 3d 00 00 00 00 7d 0b 50 b0 2d e8 ....=....}.P.-.
00000480 4c fe ff ff 58 f7 d8 3d 0a 00 00 00 0f 8c ef 00 L...X..=.....
00000490 00 00 3d 64 00 00 00 0f 8c d1 00 00 00 3d e8 03 ..=d.....=..
000004a0 00 00 0f 8c b3 00 00 00 3d 10 27 00 00 0f 8c 95 .....=.'.....
000004b0 00 00 00 3d a0 86 01 00 7c 7b 3d 40 42 0f 00 7c ...=....|{=@B...|
000004c0 61 3d 80 96 98 00 7c 47 3d 00 e1 f5 05 7c 2d 3d a=....|G=....|-=
000004d0 00 ca 9a 3b 7c 13 ba 00 00 00 00 bb 00 ca 9a 3b ...;|.....;
000004e0 f7 fb 52 e8 18 ff ff ff 58 ba 00 00 00 00 bb 00 ..R....X.....
000004f0 e1 f5 05 f7 fb 52 e8 05 ff ff ff 58 ba 00 00 00 ....R....X....
00000500 00 bb 80 96 98 00 f7 fb 52 e8 f2 fe ff ff 58 ba .....R....X.
00000510 00 00 00 00 bb 40 42 0f 00 f7 fb 52 e8 df fe ff .....@B....R....
00000520 ff 58 ba 00 00 00 00 bb a0 86 01 00 f7 fb 52 e8 .X.....R.
00000530 cc fe ff ff 58 ba 00 00 00 00 bb 10 27 00 00 f7 ....X.....'....
00000540 fb 52 e8 b9 fe ff ff 58 ba 00 00 00 00 bb e8 03 .R....X.....
00000550 00 00 f7 fb 52 e8 a6 fe ff ff 58 ba 00 00 00 00 ....R....X.....
00000560 bb 64 00 00 00 f7 fb 52 e8 93 fe ff ff 58 ba 00 .d....R....X..
00000570 00 00 00 bb 0a 00 00 00 f7 fb 52 e8 80 fe ff ff .....R.....
00000580 58 e8 7a fe ff ff c3 00 ff 15 00 10 40 00 00 00 X.z.....@...
00000590 b9 00 00 00 00 b3 03 51 53 e8 a2 fd ff ff 5b 59 .....QS.....[Y
000005a0 3c 0d 0f 84 34 01 00 00 3c 08 0f 84 94 00 00 00 <...4...<.....
000005b0 3c 2d 0f 84 09 01 00 00 3c 30 7c db 3c 39 7f d7 <-.....<0|. <9..
000005c0 2c 30 80 fb 00 74 d0 80 fb 02 75 0c 81 f9 00 00 ,0...t...u....
000005d0 00 00 75 04 3c 00 74 bf 80 fb 03 75 0a 3c 00 75 ..u.<.t...u.<.u
000005e0 04 b3 00 eb 02 b3 01 81 f9 cc cc cc 0c 7f a8 81 .....
000005f0 f9 34 33 33 f3 7c a0 88 c7 b8 0a 00 00 00 f7 e9 .433.|.....
00000600 3d 08 00 00 80 74 11 3d f8 ff ff 7f 75 13 80 ff =....t.=....u...
00000610 07 7e 0e e9 7f ff ff ff 80 ff 08 0f 8f 76 ff ff .~.....v...
00000620 ff b9 00 00 00 00 88 f9 80 fb 02 74 04 01 c1 eb .....t.....
00000630 03 29 c8 91 88 f8 51 53 e8 c3 fd ff ff 5b 59 e9 .)....QS.....[Y.
00000640 53 ff ff ff 80 fb 03 0f 84 4a ff ff ff 51 53 b0 s.....J...QS.
00000650 08 e8 7a fc ff ff b0 20 e8 73 fc ff ff b0 08 e8 ..z....s.....
00000660 6c fc ff ff 5b 59 80 fb 00 75 07 b3 03 e9 25 ff 1...[Y...u...%.
00000670 ff ff 80 fb 02 75 0f 81 f9 00 00 00 00 75 07 b3 .....u.....u..
00000680 03 e9 11 ff ff ff 89 c8 b9 0a 00 00 00 ba 00 00 .....
00000690 00 00 3d 00 00 00 00 7d 08 f7 d8 f7 f9 f7 d8 eb ..=....}.....
000006a0 02 f7 f9 89 c1 81 f9 00 00 00 00 0f 85 e6 fe ff .....
000006b0 ff 80 fb 02 0f 84 dd fe ff ff b3 03 e9 d6 fe ff .....
000006c0 ff 80 fb 03 0f 85 cd fe ff ff b0 2d 51 53 e8 fd .....-QS..
000006d0 fb ff ff 5b 59 b3 02 e9 bb fe ff ff 80 fb 03 0f ...[Y.....
000006e0 84 b2 fe ff ff 80 fb 02 75 0c 81 f9 00 00 00 00 .....u.....
000006f0 0f 84 a1 fe ff ff 51 e8 14 fd ff ff 59 89 c8 c3 .....Q....Y...
00000700 bf 7c 16 40 00 e9 24 01 00 00 8b 87 00 00 00 00 .|.@..$.
00000710 89 87 0c 00 00 00 8b 87 04 00 00 00 89 87 10 00 .....
00000720 00 00 b8 00 00 00 00 89 87 08 00 00 00 8b 87 00 .....
00000730 00 00 00 89 c3 b8 00 00 00 00 39 c3 0f 8d 0e 00 .....9.....
00000740 00 00 8b 87 0c 00 00 00 f7 d8 89 87 0c 00 00 00 .....
00000750 8b 87 04 00 00 00 89 c3 b8 00 00 00 00 39 c3 0f .....9..
00000760 8d 0e 00 00 00 8b 87 10 00 00 00 f7 d8 89 87 10 .....
00000770 00 00 00 8b 87 10 00 00 00 89 c3 b8 00 00 00 00 .....
00000780 39 c3 0f 8e 5f 00 00 00 8b 87 10 00 00 00 31 db 9..._.1.

```

00000790	43 21 d8 39 d8 0f 85 18	00 00 00 8b 87 08 00 00	C!.9.....
000007a0	00 89 c3 8b 87 0c 00 00	00 01 c3 89 d8 89 87 08
000007b0	00 00 00 8b 87 0c 00 00	00 89 c3 b8 02 00 00 00
000007c0	0f af d8 89 d8 89 87 0c	00 00 00 8b 87 10 00 00
000007d0	00 89 c3 b8 02 00 00 00	93 99 f7 fb 93 89 d8 89
000007e0	87 10 00 00 00 eb 8c 8b	87 00 00 00 00 89 c3 b8
000007f0	00 00 00 00 39 c3 0f 8d	0e 00 00 00 8b 87 08 009.....
00000800	00 00 f7 d8 89 87 08 00	00 00 8b 87 04 00 00 00
00000810	89 c3 b8 00 00 00 00 39	c3 0f 8d 0e 00 00 00 8b9.....
00000820	87 08 00 00 00 f7 d8 89	87 08 00 00 00 c3 b8 90
00000830	16 40 00 e8 a8 fb ff ff	e8 53 fd ff ff 89 87 00	.@.....S.....
00000840	00 00 00 b8 94 16 40 00	e8 93 fb ff ff e8 3e fd@.....>.
00000850	ff ff 89 87 04 00 00 00	e8 ad fe ff ff b8 98 16
00000860	40 00 e8 79 fb ff ff 8b	87 08 00 00 00 e8 ae fb	@..y.....
00000870	ff ff e8 99 fb ff ff e9	0c fd ff ff 00 00 00 00
00000880	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000890	58 3a 20 00 59 3a 20 00	58 2a 59 3d 00 00 00 00	X: .Y: .X*Y=....
000008a0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008b0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008c0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008d0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008e0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008f0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

desensamblado

```

:00401500 BF7C164000      mov edi, 0040167C
:00401505 E924010000      jmp 0040162E
:0040150A 8B8700000000      mov eax, dword ptr [edi+00000000]
:00401510 89870C000000      mov dword ptr [edi+0000000C], eax
:00401516 8B8704000000      mov eax, dword ptr [edi+00000004]
:0040151C 898710000000      mov dword ptr [edi+00000010], eax
:00401522 B800000000      mov eax, 00000000
:00401527 898708000000      mov dword ptr [edi+00000008], eax
:0040152D 8B8700000000      mov eax, dword ptr [edi+00000000]
:00401533 89C3              mov ebx, eax
:00401535 B800000000      mov eax, 00000000
:0040153A 39C3              cmp ebx, eax
:0040153C 0F8D0E000000      jnl 00401550
:00401542 8B870C000000      mov eax, dword ptr [edi+0000000C]
:00401548 F7D8              neg eax
:0040154A 89870C000000      mov dword ptr [edi+0000000C], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040153C(C)

```

|
:00401550 8B8704000000      mov eax, dword ptr [edi+00000004]
:00401556 89C3              mov ebx, eax
:00401558 B800000000      mov eax, 00000000
:0040155D 39C3              cmp ebx, eax
:0040155F 0F8D0E000000      jnl 00401573
:00401565 8B8710000000      mov eax, dword ptr [edi+00000010]
:0040156B F7D8              neg eax
:0040156D 898710000000      mov dword ptr [edi+00000010], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at
Addresses:

|:0040155F(C), :004015E5(U)

```

|
:00401573 8B8710000000      mov eax, dword ptr [edi+00000010]
:00401579 89C3              mov ebx, eax
:0040157B B800000000      mov eax, 00000000
:00401580 39C3              cmp ebx, eax
:00401582 0F8E5F000000      jle 004015E7
:00401588 8B8710000000      mov eax, dword ptr [edi+00000010]
:0040158E 31DB              xor ebx, ebx
:00401590 43              inc ebx
:00401591 21D8              and eax, ebx
:00401593 39D8              cmp eax, ebx
:00401595 0F8518000000      jne 004015B3

```

```

:0040159B 8B8708000000    mov eax, dword ptr [edi+00000008]
:004015A1 89C3               mov ebx, eax
:004015A3 8B870C000000    mov eax, dword ptr [edi+0000000C]
:004015A9 01C3               add ebx, eax
:004015AB 89D8               mov eax, ebx
:004015AD 898708000000    mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401595(C)

```

|
:004015B3 8B870C000000    mov eax, dword ptr [edi+0000000C]
:004015B9 89C3               mov ebx, eax
:004015BB B802000000    mov eax, 00000002
:004015C0 0FAFD8           imul ebx, eax
:004015C3 89D8               mov eax, ebx
:004015C5 89870C000000    mov dword ptr [edi+0000000C], eax
:004015CB 8B8710000000    mov eax, dword ptr [edi+00000010]
:004015D1 89C3               mov ebx, eax
:004015D3 B802000000    mov eax, 00000002
:004015D8 93                xchg eax, ebx
:004015D9 99                cdq
:004015DA F7FB             idiv ebx
:004015DC 93                xchg eax, ebx
:004015DD 89D8               mov eax, ebx
:004015DF 898710000000    mov dword ptr [edi+00000010], eax
:004015E5 EB8C               jmp 00401573

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401582(C)

```

|
:004015E7 8B8700000000    mov eax, dword ptr [edi+00000000]
:004015ED 89C3               mov ebx, eax
:004015EF B800000000    mov eax, 00000000
:004015F4 39C3               cmp ebx, eax
:004015F6 0F8D0E000000    jnl 0040160A
:004015FC 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401602 F7D8               neg eax
:00401604 898708000000    mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004015F6(C)

```

|
:0040160A 8B8704000000    mov eax, dword ptr [edi+00000004]
:00401610 89C3               mov ebx, eax
:00401612 B800000000    mov eax, 00000000
:00401617 39C3               cmp ebx, eax

```


03

```

var X, Y, Q, R;

procedure DIVIDIR;
var V, W;
begin
  Q := 0;
  R := X; if R < 0 then R := -R;
  W := Y; if W < 0 then W := -W;
  V := Y; if V < 0 then V := -V;
  while W <= R do W := W * 2;
  while W > V do
    begin
      Q := Q * 2; W := W / 2;
      if W <= R then
        begin
          R := R - W; Q := Q + 1
        end
      end;
    if X < 0 then R := -R;
    if X < 0 then Q := -Q;
    if Y < 0 then Q := -Q;
  end;

procedure OTRO;
  procedure DIVIDIR;
  begin
    Q := X / Y; R := X - Y * Q
  end;
  call DIVIDIR;

procedure SALIDA;
begin
  write ('Cociente: ', Q); writeln;
  write ('Resto: ', R); writeln;
end;

begin
  write ('Dividendo: '); readln (X);
  write ('Divisor: '); readln (Y);
  writeln;
  if Y <> 0 then
    begin
      write ('Metodo 1'); writeln;
      call DIVIDIR;
      call SALIDA; writeln;
    end
  end;
end;

```

```
        write ('Metodo 2'); writeln;  
        call OTRO;  
        call SALIDA;  
    end  
end.
```



```

000003d0    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
000003e0    60 89 c6 30 c0 02 06 74    08 46 e8 e1 fe ff ff eb    `..0...t.F.....
000003f0    f2 61 90 c3 00 00 00 00    00 00 00 00 00 00 00 00    .a.....
00000400    04 30 e8 c9 fe ff ff c3    00 00 00 00 00 00 00 00    .0.....
00000410    b0 0d e8 b9 fe ff ff b0    0a e8 b2 fe ff ff c3 00    .....
00000420    3d 00 00 00 80 75 4e b0    2d e8 a2 fe ff ff b0 02    =....uN.-.....
00000430    e8 cb ff ff ff b0 01 e8    c4 ff ff ff b0 04 e8 bd    .....
00000440    ff ff ff b0 07 e8 b6 ff    ff ff b0 04 e8 af ff ff    .....
00000450    ff b0 08 e8 a8 ff ff ff    b0 03 e8 a1 ff ff ff b0    .....
00000460    06 e8 9a ff ff ff b0 04    e8 93 ff ff ff b0 08 e8    .....
00000470    8c ff ff ff c3 3d 00 00    00 00 7d 0b 50 b0 2d e8    .....=....}.P.-.
00000480    4c fe ff ff 58 f7 d8 3d    0a 00 00 00 0f 8c ef 00    L...X..=.....
00000490    00 00 3d 64 00 00 00 0f    8c d1 00 00 00 3d e8 03    ..=d.....=.
000004a0    00 00 0f 8c b3 00 00 00    3d 10 27 00 00 0f 8c 95    .....=.'.....
000004b0    00 00 00 3d a0 86 01 00    7c 7b 3d 40 42 0f 00 7c    ...=....|{@B..|
000004c0    61 3d 80 96 98 00 7c 47    3d 00 e1 f5 05 7c 2d 3d    a=....|G=....|=
000004d0    00 ca 9a 3b 7c 13 ba 00    00 00 00 bb 00 ca 9a 3b    ...;|.....;
000004e0    f7 fb 52 e8 18 ff ff ff    58 ba 00 00 00 00 bb 00    ..R....X.....
000004f0    e1 f5 05 f7 fb 52 e8 05    ff ff ff 58 ba 00 00 00    .....R....X....
00000500    00 bb 80 96 98 00 f7 fb    52 e8 f2 fe ff ff 58 ba    .....R....X.
00000510    00 00 00 00 bb 40 42 0f    00 f7 fb 52 e8 df fe ff    .....@B....R....
00000520    ff 58 ba 00 00 00 00 bb    a0 86 01 00 f7 fb 52 e8    .X.....R.
00000530    cc fe ff ff 58 ba 00 00    00 00 bb 10 27 00 00 f7    ....X.....'...
00000540    fb 52 e8 b9 fe ff ff 58    ba 00 00 00 00 bb e8 03    .R....X.....
00000550    00 00 f7 fb 52 e8 a6 fe    ff ff 58 ba 00 00 00 00    ....R....X.....
00000560    bb 64 00 00 00 f7 fb 52    e8 93 fe ff ff 58 ba 00    .d.....R....X..
00000570    00 00 00 bb 0a 00 00 00    f7 fb 52 e8 80 fe ff ff    .....R.....
00000580    58 e8 7a fe ff ff c3 00    ff 15 00 10 40 00 00 00    X.z.....@...
00000590    b9 00 00 00 00 b3 03 51    53 e8 a2 fd ff ff 5b 59    .....QS.....[Y
000005a0    3c 0d 0f 84 34 01 00 00    3c 08 0f 84 94 00 00 00    <...4...<.....
000005b0    3c 2d 0f 84 09 01 00 00    3c 30 7c db 3c 39 7f d7    <-.....<0|. <9..
000005c0    2c 30 80 fb 00 74 d0 80    fb 02 75 0c 81 f9 00 00    ,0...t....u....
000005d0    00 00 75 04 3c 00 74 bf    80 fb 03 75 0a 3c 00 75    ..u.<.t....u.<.u
000005e0    04 b3 00 eb 02 b3 01 81    f9 cc cc cc 0c 7f a8 81    .....
000005f0    f9 34 33 33 f3 7c a0 88    c7 b8 0a 00 00 00 f7 e9    .433.|.....
00000600    3d 08 00 00 80 74 11 3d    f8 ff ff 7f 75 13 80 ff    =....t.=...u...
00000610    07 7e 0e e9 7f ff ff ff    80 ff 08 0f 8f 76 ff ff    .~.....v..
00000620    ff b9 00 00 00 00 88 f9    80 fb 02 74 04 01 c1 eb    .....t....
00000630    03 29 c8 91 88 f8 51 53    e8 c3 fd ff ff 5b 59 e9    .)....QS.....[Y.
00000640    53 ff ff ff 80 fb 03 0f    84 4a ff ff ff 51 53 b0    S.....J....QS.
00000650    08 e8 7a fc ff ff b0 20    e8 73 fc ff ff b0 08 e8    .z.... .s.....
00000660    6c fc ff ff 5b 59 80 fb    00 75 07 b3 03 e9 25 ff    l...[Y...u....%.
00000670    ff ff 80 fb 02 75 0f 81    f9 00 00 00 00 75 07 b3    .....u.....u..
00000680    03 e9 11 ff ff ff 89 c8    b9 0a 00 00 00 ba 00 00    .....
00000690    00 00 3d 00 00 00 00 7d    08 f7 d8 f7 f9 f7 d8 eb    ..=....}.....
000006a0    02 f7 f9 89 c1 81 f9 00    00 00 00 0f 85 e6 fe ff    .....
000006b0    ff 80 fb 02 0f 84 dd fe    ff ff b3 03 e9 d6 fe ff    .....
000006c0    ff 80 fb 03 0f 85 cd fe    ff ff b0 2d 51 53 e8 fd    .....-QS..
000006d0    fb ff ff 5b 59 b3 02 e9    bb fe ff ff 80 fb 03 0f    ...[Y.....
000006e0    84 b2 fe ff ff 80 fb 02    75 0c 81 f9 00 00 00 00    .....u.....
000006f0    0f 84 a1 fe ff ff 51 e8    14 fd ff ff 59 89 c8 c3    .....Q.....Y...
00000700    bf 0d 18 40 00 e9 5b 02    00 00 b8 00 00 00 00 89    ...@..[.....
00000710    87 08 00 00 00 8b 87 00    00 00 00 89 87 0c 00 00    .....
00000720    00 8b 87 0c 00 00 00 89    c3 b8 00 00 00 00 39 c3    .....9.
00000730    0f 8d 0e 00 00 00 8b 87    0c 00 00 00 f7 d8 89 87    .....
00000740    0c 00 00 00 8b 87 04 00    00 00 89 87 14 00 00 00    .....
00000750    8b 87 14 00 00 00 89 c3    b8 00 00 00 00 39 c3 0f    .....9..
00000760    8d 0e 00 00 00 8b 87 14    00 00 00 f7 d8 89 87 14    .....
00000770    00 00 00 8b 87 04 00 00    00 89 87 10 00 00 00 8b    .....
00000780    87 10 00 00 00 89 c3 b8    00 00 00 00 39 c3 0f 8d    .....9...
00000790    0e 00 00 00 8b 87 10 00    00 00 f7 d8 89 87 10 00    .....
000007a0    00 00 8b 87 14 00 00 00    89 c3 8b 87 0c 00 00 00    .....

```

000007b0	39 c3 0f 8f 1a 00 00 00	8b 87 14 00 00 00 89 c3	9.....
000007c0	b8 02 00 00 00 0f af d8	89 d8 89 87 14 00 00 00
000007d0	eb d0 8b 87 14 00 00 00	89 c3 8b 87 10 00 00 00
000007e0	39 c3 0f 8e 7c 00 00 00	8b 87 08 00 00 00 89 c3	9...
000007f0	b8 02 00 00 00 0f af d8	89 d8 89 87 08 00 00 00
00000800	8b 87 14 00 00 00 89 c3	b8 02 00 00 00 93 99 f7
00000810	fb 93 89 d8 89 87 14 00	00 00 8b 87 14 00 00 00
00000820	89 c3 8b 87 0c 00 00 00	39 c3 0f 8f 2f 00 00 009.../...
00000830	8b 87 0c 00 00 00 89 c3	8b 87 14 00 00 00 29 c3).
00000840	89 d8 89 87 0c 00 00 00	8b 87 08 00 00 00 89 c3
00000850	b8 01 00 00 00 01 c3 89	d8 89 87 08 00 00 00 e9
00000860	6e ff ff ff 8b 87 00 00	00 00 89 c3 b8 00 00 00	n.....
00000870	00 39 c3 0f 8d 0e 00 00	00 8b 87 0c 00 00 00 f7	.9.....
00000880	d8 89 87 0c 00 00 00 8b	87 00 00 00 00 89 c3 b8
00000890	00 00 00 00 39 c3 0f 8d	0e 00 00 00 8b 87 08 009.....
000008a0	00 00 f7 d8 89 87 08 00	00 00 8b 87 04 00 00 00
000008b0	89 c3 b8 00 00 00 00 39	c3 0f 8d 0e 00 00 00 8b9.....
000008c0	87 08 00 00 00 f7 d8 89	87 08 00 00 00 c3 e9 43C
000008d0	00 00 00 8b 87 00 00 00	00 89 c3 8b 87 04 00 00
000008e0	00 93 99 f7 fb 93 89 d8	89 87 08 00 00 00 8b 87
000008f0	00 00 00 00 89 c3 53 8b	87 04 00 00 00 89 c3 8bS.....
00000900	87 08 00 00 00 0f af d8	89 d8 5b 29 c3 89 d8 89[).....
00000910	87 0c 00 00 00 c3 e8 b8	ff ff ff c3 b8 25 18 40%.@
00000920	00 e8 ba fa ff ff 8b 87	08 00 00 00 e8 ef fa ff
00000930	ff b8 30 18 40 00 e8 a5	fa ff ff e8 d0 fa ff ff	..0.@.....
00000940	b8 31 18 40 00 e8 96 fa	ff ff 8b 87 0c 00 00 00	.1.@.....
00000950	e8 cb fa ff ff b8 39 18	40 00 e8 81 fa ff ff e89.@.....
00000960	ac fa ff ff c3 b8 3a 18	40 00 e8 71 fa ff ff e8:.@..q....
00000970	1c fc ff ff 89 87 00 00	00 00 b8 46 18 40 00 e8F.@..
00000980	5c fa ff ff e8 07 fc ff	ff 89 87 04 00 00 00 b8	\.....
00000990	50 18 40 00 e8 47 fa ff	ff e8 72 fa ff ff 8b 87	P.@..G....r.....
000009a0	04 00 00 00 89 c3 b8 00	00 00 00 39 c3 0f 84 559...U
000009b0	00 00 00 b8 51 18 40 00	e8 23 fa ff ff b8 5a 18Q.@..#....Z.
000009c0	40 00 e8 19 fa ff ff e8	44 fa ff ff e8 39 fd ff	@.....D....9..
000009d0	ff e8 46 ff ff ff b8 5b	18 40 00 e8 00 fa ff ff	..F....[.@.....
000009e0	e8 2b fa ff ff b8 5c 18	40 00 e8 f1 f9 ff ff b8	.+....\.@.....
000009f0	65 18 40 00 e8 e7 f9 ff	ff e8 12 fa ff ff e8 13	e.@.....
00000a00	ff ff ff e8 14 ff ff ff	e9 7b fb ff ff 00 00 00{.....
00000a10	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000a20	00 00 00 00 00 43 6f 63	69 65 6e 74 65 3a 20 00Cociente: .
00000a30	00 52 65 73 74 6f 3a 20	00 00 44 69 76 69 64 65	.Resto: ..Divide
00000a40	6e 64 6f 3a 20 00 44 69	76 69 73 6f 72 3a 20 00	ndo: .Divisor: .
00000a50	00 4d 65 74 6f 64 6f 20	31 00 00 00 4d 65 74 6f	.Metodo 1...Meto
00000a60	64 6f 20 32 00 00 00 00	00 00 00 00 00 00 00 00	do 2.....
00000a70	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000a80	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000a90	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000aa0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000ab0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000ac0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000ad0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000ae0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000af0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

desensamblado

```
:00401500 BF0D184000      mov edi, 0040180D
:00401505 E95B020000      jmp 00401765
```

* Referenced by a CALL at Address:

|:004017CC

|

```
:0040150A B800000000      mov eax, 00000000
:0040150F 898708000000      mov dword ptr [edi+00000008], eax
:00401515 8B8700000000      mov eax, dword ptr [edi+00000000]
:0040151B 89870C000000      mov dword ptr [edi+0000000C], eax
:00401521 8B870C000000      mov eax, dword ptr [edi+0000000C]
:00401527 89C3              mov ebx, eax
:00401529 B800000000      mov eax, 00000000
:0040152E 39C3              cmp ebx, eax
:00401530 0F8D0E000000      jnl 00401544
:00401536 8B870C000000      mov eax, dword ptr [edi+0000000C]
:0040153C F7D8              neg eax
:0040153E 89870C000000      mov dword ptr [edi+0000000C], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401530(C)

|

```
:00401544 8B8704000000      mov eax, dword ptr [edi+00000004]
:0040154A 898714000000      mov dword ptr [edi+00000014], eax
:00401550 8B8714000000      mov eax, dword ptr [edi+00000014]
:00401556 89C3              mov ebx, eax
:00401558 B800000000      mov eax, 00000000
:0040155D 39C3              cmp ebx, eax
:0040155F 0F8D0E000000      jnl 00401573
:00401565 8B8714000000      mov eax, dword ptr [edi+00000014]
:0040156B F7D8              neg eax
:0040156D 898714000000      mov dword ptr [edi+00000014], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:0040155F(C)

|

```
:00401573 8B8704000000      mov eax, dword ptr [edi+00000004]
:00401579 898710000000      mov dword ptr [edi+00000010], eax
:0040157F 8B8710000000      mov eax, dword ptr [edi+00000010]
:00401585 89C3              mov ebx, eax
:00401587 B800000000      mov eax, 00000000
:0040158C 39C3              cmp ebx, eax
:0040158E 0F8D0E000000      jnl 004015A2
:00401594 8B8710000000      mov eax, dword ptr [edi+00000010]
```

```

:0040159A F7D8          neg eax
:0040159C 898710000000       mov dword ptr [edi+00000010], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Addresses:

|:0040158E(C), :004015D0(U)

```

|
:004015A2 8B8714000000       mov eax, dword ptr [edi+00000014]
:004015A8 89C3              mov ebx, eax
:004015AA 8B870C000000       mov eax, dword ptr [edi+0000000C]
:004015B0 39C3              cmp ebx, eax
:004015B2 0F8F1A000000       jg 004015D2
:004015B8 8B8714000000       mov eax, dword ptr [edi+00000014]
:004015BE 89C3              mov ebx, eax
:004015C0 B802000000       mov eax, 00000002
:004015C5 0FAFD8            imul ebx, eax
:004015C8 89D8              mov eax, ebx
:004015CA 898714000000       mov dword ptr [edi+00000014], eax
:004015D0 EBD0              jmp 004015A2

```

* Referenced by a (U)nconditional or (C)onditional Jump at Addresses:

|:004015B2(C), :0040165F(U)

```

|
:004015D2 8B8714000000       mov eax, dword ptr [edi+00000014]
:004015D8 89C3              mov ebx, eax
:004015DA 8B8710000000       mov eax, dword ptr [edi+00000010]
:004015E0 39C3              cmp ebx, eax
:004015E2 0F8E7C000000       jle 00401664
:004015E8 8B8708000000       mov eax, dword ptr [edi+00000008]
:004015EE 89C3              mov ebx, eax
:004015F0 B802000000       mov eax, 00000002
:004015F5 0FAFD8            imul ebx, eax
:004015F8 89D8              mov eax, ebx
:004015FA 898708000000       mov dword ptr [edi+00000008], eax
:00401600 8B8714000000       mov eax, dword ptr [edi+00000014]
:00401606 89C3              mov ebx, eax
:00401608 B802000000       mov eax, 00000002
:0040160D 93                xchg eax,ebx
:0040160E 99                cdq
:0040160F F7FB            idiv ebx
:00401611 93                xchg eax,ebx
:00401612 89D8              mov eax, ebx
:00401614 898714000000       mov dword ptr [edi+00000014], eax
:0040161A 8B8714000000       mov eax, dword ptr [edi+00000014]
:00401620 89C3              mov ebx, eax

```

```

:00401622 8B870C000000    mov eax, dword ptr [edi+0000000C]
:00401628 39C3                cmp ebx, eax
:0040162A 0F8F2F000000    jg 0040165F
:00401630 8B870C000000    mov eax, dword ptr [edi+0000000C]
:00401636 89C3                mov ebx, eax
:00401638 8B8714000000    mov eax, dword ptr [edi+00000014]
:0040163E 29C3                sub ebx, eax
:00401640 89D8                mov eax, ebx
:00401642 89870C000000    mov dword ptr [edi+0000000C], eax
:00401648 8B8708000000    mov eax, dword ptr [edi+00000008]
:0040164E 89C3                mov ebx, eax
:00401650 B801000000    mov eax, 00000001
:00401655 01C3                add ebx, eax
:00401657 89D8                mov eax, ebx
:00401659 898708000000    mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040162A(C)

```

|
:0040165F E96EFFFFFF      jmp 004015D2

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004015E2(C)

```

|
:00401664 8B8700000000    mov eax, dword ptr [edi+00000000]
:0040166A 89C3                mov ebx, eax
:0040166C B800000000    mov eax, 00000000
:00401671 39C3                cmp ebx, eax
:00401673 0F8D0E000000    jnl 00401687
:00401679 8B870C000000    mov eax, dword ptr [edi+0000000C]
:0040167F F7D8                neg eax
:00401681 89870C000000    mov dword ptr [edi+0000000C], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401673(C)

```

|
:00401687 8B8700000000    mov eax, dword ptr [edi+00000000]
:0040168D 89C3                mov ebx, eax
:0040168F B800000000    mov eax, 00000000
:00401694 39C3                cmp ebx, eax
:00401696 0F8D0E000000    jnl 004016AA
:0040169C 8B8708000000    mov eax, dword ptr [edi+00000008]
:004016A2 F7D8                neg eax
:004016A4 898708000000    mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

```

|:00401696 (C)
|
:004016AA 8B8704000000    mov eax, dword ptr [edi+00000004]
:004016B0 89C3              mov ebx, eax
:004016B2 B800000000        mov eax, 00000000
:004016B7 39C3              cmp ebx, eax
:004016B9 0F8D0E000000        jnl 004016CD
:004016BF 8B8708000000        mov eax, dword ptr [edi+00000008]
:004016C5 F7D8              neg eax
:004016C7 898708000000        mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

```

|:004016B9 (C)
|
:004016CD C3              ret
:004016CE E943000000        jmp 00401716

```

* Referenced by a CALL at Address:

```

|:00401716
|
:004016D3 8B8700000000        mov eax, dword ptr [edi+00000000]
:004016D9 89C3              mov ebx, eax
:004016DB 8B8704000000        mov eax, dword ptr [edi+00000004]
:004016E1 93              xchg eax, ebx
:004016E2 99              cdq
:004016E3 F7FB              idiv ebx
:004016E5 93              xchg eax, ebx
:004016E6 89D8              mov eax, ebx
:004016E8 898708000000        mov dword ptr [edi+00000008], eax
:004016EE 8B8700000000        mov eax, dword ptr [edi+00000000]
:004016F4 89C3              mov ebx, eax
:004016F6 53              push ebx
:004016F7 8B8704000000        mov eax, dword ptr [edi+00000004]
:004016FD 89C3              mov ebx, eax
:004016FF 8B8708000000        mov eax, dword ptr [edi+00000008]
:00401705 0FAFD8           imul ebx, eax
:00401708 89D8              mov eax, ebx
:0040170A 5B              pop ebx
:0040170B 29C3              sub ebx, eax
:0040170D 89D8              mov eax, ebx
:0040170F 89870C000000        mov dword ptr [edi+0000000C], eax
:00401715 C3              ret

```

* Referenced by a CALL at Address:

|:004017FE

|

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:004016CE(U)

|

```
:00401716 E8B8FFFFFF      call 004016D3
:0040171B C3              ret
```

```
:0040171C B825184000      mov eax, 00401825
:00401721 E8BAFAFFFF      call 004011E0
:00401726 8B8708000000  mov eax, dword ptr [edi+00000008]
:0040172C E8EFAFAFFFF      call 00401220
:00401731 B830184000      mov eax, 00401830
:00401736 E8A5FAFFFF      call 004011E0
:0040173B E8D0FAFFFF      call 00401210
:00401740 B831184000      mov eax, 00401831
:00401745 E896FAFFFF      call 004011E0
:0040174A 8B870C000000    mov eax, dword ptr [edi+0000000C]
:00401750 E8CBFAFFFF      call 00401220
:00401755 B839184000      mov eax, 00401839
:0040175A E881FAFFFF      call 004011E0
:0040175F E8ACFAFFFF      call 00401210
:00401764 C3              ret
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401505(U)

|

```
:00401765 B83A184000      mov eax, 0040183A
:0040176A E871FAFFFF      call 004011E0
:0040176F E81CFCFFFF      call 00401390
:00401774 898700000000    mov dword ptr [edi+00000000], eax
:0040177A B846184000      mov eax, 00401846
:0040177F E85CFAFFFF      call 004011E0
:00401784 E807FCFFFF      call 00401390
:00401789 898704000000    mov dword ptr [edi+00000004], eax
:0040178F B850184000      mov eax, 00401850
:00401794 E847FAFFFF      call 004011E0
:00401799 E872FAFFFF      call 00401210
:0040179E 8B8704000000    mov eax, dword ptr [edi+00000004]
:004017A4 89C3              mov ebx, eax
:004017A6 B800000000      mov eax, 00000000
:004017AB 39C3              cmp ebx, eax
```

```

:004017AD 0F8455000000      je 00401808
:004017B3 B851184000      mov eax, 00401851
:004017B8 E823FAFFFF      call 004011E0
:004017BD B85A184000      mov eax, 0040185A
:004017C2 E819FAFFFF      call 004011E0
:004017C7 E844FAFFFF      call 00401210
:004017CC E839FDFFFF      call 0040150A
:004017D1 E846FFFFFF      call 0040171C
:004017D6 B85B184000      mov eax, 0040185B
:004017DB E800FAFFFF      call 004011E0
:004017E0 E82BFAFFFF      call 00401210
:004017E5 B85C184000      mov eax, 0040185C
:004017EA E8F1F9FFFF      call 004011E0
:004017EF B865184000      mov eax, 00401865
:004017F4 E8E7F9FFFF      call 004011E0
:004017F9 E812FAFFFF      call 00401210
:004017FE E813FFFFFF      call 00401716
:00401803 E814FFFFFF      call 0040171C

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

| :004017AD(C)

|

```

:00401808 E97BFBFFFF      jmp 00401388
:0040180D 00000000000000000000  BYTE 10 DUP(0)
:00401817 00000000000000000000  BYTE 10 DUP(0)
:00401821 00000000      BYTE 4 DUP(0)

```

```

:00401825 43      inc ebx
:00401826 6F      outsd
:00401827 636965  arpl dword ptr [ecx+65], ebp
:0040182A 6E      outsb
:0040182B 7465      je 00401892
:0040182D 3A20      cmp ah, byte ptr [eax]
:0040182F 0000      add byte ptr [eax], al
:00401831 52      push edx
:00401832 65      BYTE 065h

```

```

:00401833 7374      jnb 004018A9
:00401835 6F      outsd
:00401836 3A20      cmp ah, byte ptr [eax]
:00401838 0000      add byte ptr [eax], al
:0040183A 44      inc esp
:0040183B 69766964656E64  imul esi, dword ptr [esi+69],
646E6564

```



```

:00401842 6F                                outsd
:00401843 3A20                             cmp ah, byte ptr [eax]
:00401845 00446976                         add byte ptr [ecx+2*ebp+76], al
:00401849 69736F723A2000                   imul esi, dword ptr [ebx+6F],
00203A72
:00401850 004D65                         add byte ptr [ebp+65], cl
:00401853 746F                             je 004018C4
:00401855 64                             BYTE 064h

:00401856 6F                                outsd
:00401857 2031                             and byte ptr [ecx], dh
:00401859 000000                           BYTE 3 DUP(0)

:0040185C 4D                                dec ebp
:0040185D 65                             BYTE 065h

:0040185E 746F                             je 004018CF
:00401860 64                             BYTE 064h

:00401861 6F                                outsd
:00401862 2032                             and byte ptr [edx], dh
:00401864 0000000000000000000000000000    BYTE 10 DUP(0)

```

04

```
var X, Y, Z;

procedure MCD;
var F,G;
begin
  F := X; G := Y;
  while F <> G do
    begin
      if F < G then G := G - F;
      IF G < F then F := F - G
    end;
  Z:= F
end;

begin
  write ('X: '); readln (X);
  if X > 0 then
    begin
      write ('Y: '); readln (Y);
      if Y > 0 then
        begin
          call MCD;
          writeln ('MCD: ', Z); writeln
        end
      end
    end
end.
```



```

000003d0    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
000003e0    60 89 c6 30 c0 02 06 74    08 46 e8 e1 fe ff ff eb    `..0...t.F.....
000003f0    f2 61 90 c3 00 00 00 00    00 00 00 00 00 00 00 00    .a.....
00000400    04 30 e8 c9 fe ff ff c3    00 00 00 00 00 00 00 00    .0.....
00000410    b0 0d e8 b9 fe ff ff b0    0a e8 b2 fe ff ff c3 00    .....
00000420    3d 00 00 00 80 75 4e b0    2d e8 a2 fe ff ff b0 02    =....uN.-.....
00000430    e8 cb ff ff ff b0 01 e8    c4 ff ff ff b0 04 e8 bd    .....
00000440    ff ff ff b0 07 e8 b6 ff    ff ff b0 04 e8 af ff ff    .....
00000450    ff b0 08 e8 a8 ff ff ff    b0 03 e8 a1 ff ff ff b0    .....
00000460    06 e8 9a ff ff ff b0 04    e8 93 ff ff ff b0 08 e8    .....
00000470    8c ff ff ff c3 3d 00 00    00 00 7d 0b 50 b0 2d e8    .....=....}.P.-.
00000480    4c fe ff ff 58 f7 d8 3d    0a 00 00 00 0f 8c ef 00    L...X..=.....
00000490    00 00 3d 64 00 00 00 0f    8c d1 00 00 00 3d e8 03    ..=d.....=.
000004a0    00 00 0f 8c b3 00 00 00    3d 10 27 00 00 0f 8c 95    .....=.'.....
000004b0    00 00 00 3d a0 86 01 00    7c 7b 3d 40 42 0f 00 7c    ...=....|{@B..|
000004c0    61 3d 80 96 98 00 7c 47    3d 00 e1 f5 05 7c 2d 3d    a=....|G=....|=
000004d0    00 ca 9a 3b 7c 13 ba 00    00 00 00 bb 00 ca 9a 3b    ...;|.....;
000004e0    f7 fb 52 e8 18 ff ff ff    58 ba 00 00 00 00 bb 00    ..R....X.....
000004f0    e1 f5 05 f7 fb 52 e8 05    ff ff ff 58 ba 00 00 00    .....R....X....
00000500    00 bb 80 96 98 00 f7 fb    52 e8 f2 fe ff ff 58 ba    .....R....X.
00000510    00 00 00 00 bb 40 42 0f    00 f7 fb 52 e8 df fe ff    .....@B....R....
00000520    ff 58 ba 00 00 00 00 bb    a0 86 01 00 f7 fb 52 e8    .X.....R.
00000530    cc fe ff ff 58 ba 00 00    00 00 bb 10 27 00 00 f7    ....X.....'...
00000540    fb 52 e8 b9 fe ff ff 58    ba 00 00 00 00 bb e8 03    .R.....X.....
00000550    00 00 f7 fb 52 e8 a6 fe    ff ff 58 ba 00 00 00 00    ....R....X.....
00000560    bb 64 00 00 00 f7 fb 52    e8 93 fe ff ff 58 ba 00    .d.....R....X..
00000570    00 00 00 bb 0a 00 00 00    f7 fb 52 e8 80 fe ff ff    .....R.....
00000580    58 e8 7a fe ff ff c3 00    ff 15 00 10 40 00 00 00    X.z.....@...
00000590    b9 00 00 00 00 b3 03 51    53 e8 a2 fd ff ff 5b 59    .....QS.....[Y
000005a0    3c 0d 0f 84 34 01 00 00    3c 08 0f 84 94 00 00 00    <...4...<.....
000005b0    3c 2d 0f 84 09 01 00 00    3c 30 7c db 3c 39 7f d7    <-.....<0|.<9..
000005c0    2c 30 80 fb 00 74 d0 80    fb 02 75 0c 81 f9 00 00    ,0...t....u.....
000005d0    00 00 75 04 3c 00 74 bf    80 fb 03 75 0a 3c 00 75    ..u.<.t....u.<.u
000005e0    04 b3 00 eb 02 b3 01 81    f9 cc cc cc 0c 7f a8 81    .....
000005f0    f9 34 33 33 f3 7c a0 88    c7 b8 0a 00 00 00 f7 e9    .433.|.....
00000600    3d 08 00 00 80 74 11 3d    f8 ff ff 7f 75 13 80 ff    =....t.=....u...
00000610    07 7e 0e e9 7f ff ff ff    80 ff 08 0f 8f 76 ff ff    .~.....v...
00000620    ff b9 00 00 00 00 88 f9    80 fb 02 74 04 01 c1 eb    .....t....
00000630    03 29 c8 91 88 f8 51 53    e8 c3 fd ff ff 5b 59 e9    .)....QS.....[Y.
00000640    53 ff ff ff 80 fb 03 0f    84 4a ff ff ff 51 53 b0    S.....J...QS.
00000650    08 e8 7a fc ff ff b0 20    e8 73 fc ff ff b0 08 e8    ..z....s.....
00000660    6c fc ff ff 5b 59 80 fb    00 75 07 b3 03 e9 25 ff    l...[Y...u...%.
00000670    ff ff 80 fb 02 75 0f 81    f9 00 00 00 00 75 07 b3    .....u.....u..
00000680    03 e9 11 ff ff ff 89 c8    b9 0a 00 00 00 ba 00 00    .....
00000690    00 00 3d 00 00 00 00 7d    08 f7 d8 f7 f9 f7 d8 eb    ..=....}).....
000006a0    02 f7 f9 89 c1 81 f9 00    00 00 00 0f 85 e6 fe ff    .....
000006b0    ff 80 fb 02 0f 84 dd fe    ff ff b3 03 e9 d6 fe ff    .....
000006c0    ff 80 fb 03 0f 85 cd fe    ff ff b0 2d 51 53 e8 fd    .....-QS..
000006d0    fb ff ff 5b 59 b3 02 e9    bb fe ff ff 80 fb 03 0f    ...[Y.....
000006e0    84 b2 fe ff ff 80 fb 02    75 0c 81 f9 00 00 00 00    .....u.....
000006f0    0f 84 a1 fe ff ff 51 e8    14 fd ff ff 59 89 c8 c3    .....Q.....Y...
00000700    bf 2a 16 40 00 e9 99 00    00 00 8b 87 00 00 00 00    .*.@.....
00000710    89 87 0c 00 00 00 8b 87    04 00 00 00 89 87 10 00    .....
00000720    00 00 8b 87 0c 00 00 00    89 c3 8b 87 10 00 00 00    .....
00000730    39 c3 0f 84 5e 00 00 00    8b 87 0c 00 00 00 89 c3    9...^.....
00000740    8b 87 10 00 00 00 39 c3    0f 8d 18 00 00 00 8b 87    .....9.....
00000750    10 00 00 00 89 c3 8b 87    0c 00 00 00 29 c3 89 d8    .....
00000760    89 87 10 00 00 00 8b 87    10 00 00 00 89 c3 8b 87    .....
00000770    0c 00 00 00 39 c3 0f 8d    18 00 00 00 8b 87 0c 00    ....9.....
00000780    00 00 89 c3 8b 87 10 00    00 00 29 c3 89 d8 89 87    .....
00000790    0c 00 00 00 eb 8c 8b 87    0c 00 00 00 89 87 08 00    .....
000007a0    00 00 c3 b8 3e 16 40 00    e8 33 fc ff ff e8 de fd    ....>.@..3.....

```

000007b0	ff ff 89 87 00 00 00 00	8b 87 00 00 00 00 89 c3
000007c0	b8 00 00 00 00 39 c3 0f	8e 58 00 00 00 b8 42 169...X...B.
000007d0	40 00 e8 09 fc ff ff e8	b4 fd ff ff 89 87 04 00	@.....
000007e0	00 00 8b 87 04 00 00 00	89 c3 b8 00 00 00 00 399
000007f0	c3 0f 8e 2e 00 00 00 e8	0e ff ff ff b8 46 16 40F.@
00000800	00 e8 da fb ff ff 8b 87	08 00 00 00 e8 0f fc ff
00000810	ff e8 fa fb ff ff b8 4c	16 40 00 e8 c0 fb ff ffL.@.....
00000820	e8 eb fb ff ff e9 5e fd	ff ff 00 00 00 00 00 00^.....
00000830	00 00 00 00 00 00 00 00	00 00 00 00 00 00 58 3aX:
00000840	20 00 59 3a 20 00 4d 43	44 3a 20 00 00 00 00 00	.Y: .MCD:
00000850	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000860	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000870	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000880	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000890	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008a0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008b0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008c0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008d0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008e0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008f0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

desensamblado

```

:00401500 BF2A164000      mov edi, 0040162A
:00401505 E999000000      jmp 004015A3
:0040150A 8B8700000000      mov eax, dword ptr [edi+00000000]
:00401510 89870C000000      mov dword ptr [edi+0000000C], eax
:00401516 8B8704000000      mov eax, dword ptr [edi+00000004]
:0040151C 898710000000      mov dword ptr [edi+00000010], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401594(U)

```

|
:00401522 8B870C000000      mov eax, dword ptr [edi+0000000C]
:00401528 89C3              mov ebx, eax
:0040152A 8B8710000000      mov eax, dword ptr [edi+00000010]
:00401530 39C3              cmp ebx, eax
:00401532 0F845E000000      je 00401596
:00401538 8B870C000000      mov eax, dword ptr [edi+0000000C]
:0040153E 89C3              mov ebx, eax
:00401540 8B8710000000      mov eax, dword ptr [edi+00000010]
:00401546 39C3              cmp ebx, eax
:00401548 0F8D18000000      jnl 00401566
:0040154E 8B8710000000      mov eax, dword ptr [edi+00000010]
:00401554 89C3              mov ebx, eax
:00401556 8B870C000000      mov eax, dword ptr [edi+0000000C]
:0040155C 29C3              sub ebx, eax
:0040155E 89D8              mov eax, ebx
:00401560 898710000000      mov dword ptr [edi+00000010], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401548(C)

```

|
:00401566 8B8710000000      mov eax, dword ptr [edi+00000010]
:0040156C 89C3              mov ebx, eax
:0040156E 8B870C000000      mov eax, dword ptr [edi+0000000C]
:00401574 39C3              cmp ebx, eax
:00401576 0F8D18000000      jnl 00401594
:0040157C 8B870C000000      mov eax, dword ptr [edi+0000000C]
:00401582 89C3              mov ebx, eax
:00401584 8B8710000000      mov eax, dword ptr [edi+00000010]
:0040158A 29C3              sub ebx, eax
:0040158C 89D8              mov eax, ebx
:0040158E 89870C000000      mov dword ptr [edi+0000000C], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401576(C)

```
|
:00401594 EB8C jmp 00401522
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401532(C)

```
|
:00401596 8B870C000000 mov eax, dword ptr [edi+0000000C]
:0040159C 898708000000 mov dword ptr [edi+00000008], eax
:004015A2 C3 ret
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401505(U)

```
|
:004015A3 B83E164000 mov eax, 0040163E
:004015A8 E833FCFFFF call 004011E0
:004015AD E8DEFDFFFF call 00401390
:004015B2 898700000000 mov dword ptr [edi+00000000], eax
:004015B8 8B8700000000 mov eax, dword ptr [edi+00000000]
:004015BE 89C3 mov ebx, eax
:004015C0 B800000000 mov eax, 00000000
:004015C5 39C3 cmp ebx, eax
:004015C7 0F8E58000000 jle 00401625
:004015CD B842164000 mov eax, 00401642
:004015D2 E809FCFFFF call 004011E0
:004015D7 E8B4FDFFFF call 00401390
:004015DC 898704000000 mov dword ptr [edi+00000004], eax
:004015E2 8B8704000000 mov eax, dword ptr [edi+00000004]
:004015E8 89C3 mov ebx, eax
:004015EA B800000000 mov eax, 00000000
:004015EF 39C3 cmp ebx, eax
:004015F1 0F8E2E000000 jle 00401625
:004015F7 E80EFFFFFF call 0040150A
:004015FC B846164000 mov eax, 00401646
:00401601 E8DAFBFFFF call 004011E0
:00401606 8B8708000000 mov eax, dword ptr [edi+00000008]
:0040160C E80FFCFFFF call 00401220
:00401611 E8FAFBFFFF call 00401210
:00401616 B84C164000 mov eax, 0040164C
:0040161B E8C0FBFFFF call 004011E0
:00401620 E8EBFBFFFF call 00401210
```

* Referenced by a (U)nconditional or (C)onditional Jump at
Addresses:

|:004015C7(C), :004015F1(C)

```
|
:00401625 E95EFDFFFF      jmp 00401388
:0040162A 00000000000000000000  BYTE 10 DUP(0)
:00401634 00000000000000000000  BYTE 10 DUP(0)


:0040163E 58              pop eax
:0040163F 3A20              cmp ah, byte ptr [eax]
:00401641 00593A          add byte ptr [ecx+3A], bl
:00401644 2000              and byte ptr [eax], al
:00401646 4D              dec ebp
:00401647 43              inc ebx
:00401648 44              inc esp
:00401649 3A20              cmp ah, byte ptr [eax]
:0040164B 00000000000000000000  BYTE 10 DUP(0)
```


05

```
VAR R, N;
```

```
PROCEDURE INICIALIZAR;
```

```
CONST UNO = 1;
```

```
R := -(-UNO);
```

```
PROCEDURE RAIZ;
```

```
BEGIN
```

```
    CALL INICIALIZAR;
```

```
    WHILE R * R < N DO R := R + 1
```

```
END;
```

```
BEGIN
```

```
    WRITE ('N: '); READLN (N);
```

```
    WRITE ('RAIZ CUADRADA DE ', N, ': ');
```

```
    IF N < 0 THEN WRITE ('ERROR');
```

```
    IF N = 0 THEN WRITE (0);
```

```
    IF N > 0 THEN
```

```
        BEGIN
```

```
            CALL RAIZ;
```

```
            IF R*R<>N THEN WRITE (R - 1, '..');
```

```
            WRITE (R);
```

```
        END;
```

```
    WRITELN
```

```
END.
```



```

000003d0    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
000003e0    60 89 c6 30 c0 02 06 74    08 46 e8 e1 fe ff ff eb    `..0...t.F.....
000003f0    f2 61 90 c3 00 00 00 00    00 00 00 00 00 00 00 00    .a.....
00000400    04 30 e8 c9 fe ff ff c3    00 00 00 00 00 00 00 00    .0.....
00000410    b0 0d e8 b9 fe ff ff b0    0a e8 b2 fe ff ff c3 00    .....
00000420    3d 00 00 00 80 75 4e b0    2d e8 a2 fe ff ff b0 02    =....uN.-.....
00000430    e8 cb ff ff ff b0 01 e8    c4 ff ff ff b0 04 e8 bd    .....
00000440    ff ff ff b0 07 e8 b6 ff    ff ff b0 04 e8 af ff ff    .....
00000450    ff b0 08 e8 a8 ff ff ff    b0 03 e8 a1 ff ff ff b0    .....
00000460    06 e8 9a ff ff ff b0 04    e8 93 ff ff ff b0 08 e8    .....
00000470    8c ff ff ff c3 3d 00 00    00 00 7d 0b 50 b0 2d e8    .....=....}.P.-.
00000480    4c fe ff ff 58 f7 d8 3d    0a 00 00 00 0f 8c ef 00    L...X..=.....
00000490    00 00 3d 64 00 00 00 0f    8c d1 00 00 00 3d e8 03    ..=d.....=.
000004a0    00 00 0f 8c b3 00 00 00    3d 10 27 00 00 0f 8c 95    .....=.'.....
000004b0    00 00 00 3d a0 86 01 00    7c 7b 3d 40 42 0f 00 7c    ...=....|{@B..|
000004c0    61 3d 80 96 98 00 7c 47    3d 00 e1 f5 05 7c 2d 3d    a=....|G=....|=
000004d0    00 ca 9a 3b 7c 13 ba 00    00 00 00 bb 00 ca 9a 3b    ...;|.....;
000004e0    f7 fb 52 e8 18 ff ff ff    58 ba 00 00 00 00 bb 00    ..R....X.....
000004f0    e1 f5 05 f7 fb 52 e8 05    ff ff ff 58 ba 00 00 00    .....R....X....
00000500    00 bb 80 96 98 00 f7 fb    52 e8 f2 fe ff ff 58 ba    .....R....X.
00000510    00 00 00 00 bb 40 42 0f    00 f7 fb 52 e8 df fe ff    .....@B....R....
00000520    ff 58 ba 00 00 00 00 bb    a0 86 01 00 f7 fb 52 e8    .X.....R.
00000530    cc fe ff ff 58 ba 00 00    00 00 bb 10 27 00 00 f7    ....X.....'...
00000540    fb 52 e8 b9 fe ff ff 58    ba 00 00 00 00 bb e8 03    .R.....X.....
00000550    00 00 f7 fb 52 e8 a6 fe    ff ff 58 ba 00 00 00 00    ....R....X.....
00000560    bb 64 00 00 00 f7 fb 52    e8 93 fe ff ff 58 ba 00    .d.....R....X..
00000570    00 00 00 bb 0a 00 00 00    f7 fb 52 e8 80 fe ff ff    .....R.....
00000580    58 e8 7a fe ff ff c3 00    ff 15 00 10 40 00 00 00    X.z.....@...
00000590    b9 00 00 00 00 b3 03 51    53 e8 a2 fd ff ff 5b 59    .....QS.....[Y
000005a0    3c 0d 0f 84 34 01 00 00    3c 08 0f 84 94 00 00 00    <...4...<.....
000005b0    3c 2d 0f 84 09 01 00 00    3c 30 7c db 3c 39 7f d7    <-.....<0|.<9..
000005c0    2c 30 80 fb 00 74 d0 80    fb 02 75 0c 81 f9 00 00    ,0...t....u.....
000005d0    00 00 75 04 3c 00 74 bf    80 fb 03 75 0a 3c 00 75    ..u.<.t....u.<.u
000005e0    04 b3 00 eb 02 b3 01 81    f9 cc cc cc 0c 7f a8 81    .....
000005f0    f9 34 33 33 f3 7c a0 88    c7 b8 0a 00 00 00 f7 e9    .433.|.....
00000600    3d 08 00 00 80 74 11 3d    f8 ff ff 7f 75 13 80 ff    =....t.=....u...
00000610    07 7e 0e e9 7f ff ff ff    80 ff 08 0f 8f 76 ff ff    .~.....v...
00000620    ff b9 00 00 00 00 88 f9    80 fb 02 74 04 01 c1 eb    .....t....
00000630    03 29 c8 91 88 f8 51 53    e8 c3 fd ff ff 5b 59 e9    .)....QS.....[Y.
00000640    53 ff ff ff 80 fb 03 0f    84 4a ff ff ff 51 53 b0    S.....J...QS.
00000650    08 e8 7a fc ff ff b0 20    e8 73 fc ff ff b0 08 e8    ..z....s.....
00000660    6c fc ff ff 5b 59 80 fb    00 75 07 b3 03 e9 25 ff    l...[Y...u...%.
00000670    ff ff 80 fb 02 75 0f 81    f9 00 00 00 00 75 07 b3    .....u.....u..
00000680    03 e9 11 ff ff ff 89 c8    b9 0a 00 00 00 ba 00 00    .....
00000690    00 00 3d 00 00 00 00 7d    08 f7 d8 f7 f9 f7 d8 eb    ..=....}.....
000006a0    02 f7 f9 89 c1 81 f9 00    00 00 00 0f 85 e6 fe ff    .....
000006b0    ff 80 fb 02 0f 84 dd fe    ff ff b3 03 e9 d6 fe ff    .....
000006c0    ff 80 fb 03 0f 85 cd fe    ff ff b0 2d 51 53 e8 fd    .....-QS..
000006d0    fb ff ff 5b 59 b3 02 e9    bb fe ff ff 80 fb 03 0f    ...[Y.....
000006e0    84 b2 fe ff ff 80 fb 02    75 0c 81 f9 00 00 00 00    .....u.....
000006f0    0f 84 a1 fe ff ff 51 e8    14 fd ff ff 59 89 c8 c3    .....Q.....Y...
00000700    bf 4a 16 40 00 e9 52 00    00 00 b8 01 00 00 00 f7    .J.@..R.....
00000710    d8 f7 d8 89 87 00 00 00    00 c3 e8 eb ff ff ff 8b    .....
00000720    87 00 00 00 00 89 c3 8b    87 00 00 00 00 0f af d8    .....
00000730    89 d8 89 c3 8b 87 04 00    00 00 39 c3 0f 8d 19 00    .....9.....
00000740    00 00 8b 87 00 00 00 00    89 c3 b8 01 00 00 00 01    .....
00000750    c3 89 d8 89 87 00 00 00    00 eb c4 c3 b8 52 16 40    .....R.@
00000760    00 e8 7a fc ff ff e8 25    fe ff ff 89 87 04 00 00    ..z....%.
00000770    00 b8 56 16 40 00 e8 65    fc ff ff 8b 87 04 00 00    ..V.@..e.....
00000780    00 e8 9a fc ff ff b8 68    16 40 00 e8 50 fc ff ff    .....h.@..P...
00000790    8b 87 04 00 00 00 89 c3    b8 00 00 00 00 39 c3 0f    .....9...
000007a0    8d 0a 00 00 00 b8 6b 16    40 00 e8 31 fc ff ff 8b    .....k.@..1....

```

000007b0	87 04 00 00 00 89 c3 b8	00 00 00 00 39 c3 0f 859...
000007c0	0a 00 00 00 b8 00 00 00	00 e8 52 fc ff ff 8b 87R.....
000007d0	04 00 00 00 89 c3 b8 00	00 00 00 39 c3 0f 8e 539...S
000007e0	00 00 00 e8 32 ff ff ff	8b 87 00 00 00 00 89 c32.....
000007f0	8b 87 00 00 00 00 0f af	d8 89 d8 89 c3 8b 87 04
00000800	00 00 00 39 c3 0f 84 20	00 00 00 8b 87 00 00 00	...9... ..
00000810	00 89 c3 b8 01 00 00 00	29 c3 89 d8 e8 ff fb ff).....
00000820	ff b8 71 16 40 00 e8 b5	fb ff ff 8b 87 00 00 00	..q.@.....
00000830	00 e8 ea fb ff ff b8 74	16 40 00 e8 a0 fb ff fft.@.....
00000840	e8 cb fb ff ff e9 3e fd	ff ff 00 00 00 00 00 00>.....
00000850	00 00 4e 3a 20 00 52 41	49 5a 20 43 55 41 44 52	..N: .RAIZ CUADR
00000860	41 44 41 20 44 45 20 00	3a 20 00 45 52 52 4f 52	ADA DE .: .ERROR
00000870	00 2e 2e 00 00 00 00 00	00 00 00 00 00 00 00 00
00000880	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000890	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008a0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008b0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008c0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008d0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008e0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008f0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

desensamblado

```
:00401500 BF4A164000      mov edi, 0040164A
:00401505 E952000000      jmp 0040155C
```

* Referenced by a CALL at Address:

|:0040151A

|

```
:0040150A B801000000      mov eax, 00000001
:0040150F F7D8          neg eax
:00401511 F7D8          neg eax
:00401513 898700000000      mov dword ptr [edi+00000000], eax
:00401519 C3              ret
```

```
:0040151A E8EBFFFFFF      call 0040150A
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401559(U)

|

```
:0040151F 8B8700000000      mov eax, dword ptr [edi+00000000]
:00401525 89C3              mov ebx, eax
:00401527 8B8700000000      mov eax, dword ptr [edi+00000000]
:0040152D 0FAFD8            imul ebx, eax
:00401530 89D8              mov eax, ebx
:00401532 89C3              mov ebx, eax
:00401534 8B8704000000      mov eax, dword ptr [edi+00000004]
:0040153A 39C3              cmp ebx, eax
:0040153C 0F8D19000000      jnl 0040155B
:00401542 8B8700000000      mov eax, dword ptr [edi+00000000]
:00401548 89C3              mov ebx, eax
:0040154A B801000000      mov eax, 00000001
:0040154F 01C3              add ebx, eax
:00401551 89D8              mov eax, ebx
:00401553 898700000000      mov dword ptr [edi+00000000], eax
:00401559 EBC4              jmp 0040151F
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:0040153C(C)

|

```
:0040155B C3              ret
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401505(U)

|

```
:0040155C B852164000      mov eax, 00401652
```

```

:00401561 E87AFCFFFF      call 004011E0
:00401566 E825FEFFFF      call 00401390
:0040156B 898704000000    mov dword ptr [edi+00000004], eax
:00401571 B856164000      mov eax, 00401656
:00401576 E865FCFFFF      call 004011E0
:0040157B 8B8704000000    mov eax, dword ptr [edi+00000004]
:00401581 E89AFCFFFF      call 00401220
:00401586 B868164000      mov eax, 00401668
:0040158B E850FCFFFF      call 004011E0
:00401590 8B8704000000    mov eax, dword ptr [edi+00000004]
:00401596 89C3          mov ebx, eax
:00401598 B800000000      mov eax, 00000000
:0040159D 39C3          cmp ebx, eax
:0040159F 0F8D0A000000    jnl 004015AF
:004015A5 B86B164000      mov eax, 0040166B
:004015AA E831FCFFFF      call 004011E0

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

| :0040159F(C)

|

```

:004015AF 8B8704000000    mov eax, dword ptr [edi+00000004]
:004015B5 89C3          mov ebx, eax
:004015B7 B800000000      mov eax, 00000000
:004015BC 39C3          cmp ebx, eax
:004015BE 0F850A000000    jne 004015CE
:004015C4 B800000000      mov eax, 00000000
:004015C9 E852FCFFFF      call 00401220

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

| :004015BE(C)

|

```

:004015CE 8B8704000000    mov eax, dword ptr [edi+00000004]
:004015D4 89C3          mov ebx, eax
:004015D6 B800000000      mov eax, 00000000
:004015DB 39C3          cmp ebx, eax
:004015DD 0F8E53000000    jle 00401636
:004015E3 E832FFFFFF      call 0040151A
:004015E8 8B8700000000    mov eax, dword ptr [edi+00000000]
:004015EE 89C3          mov ebx, eax
:004015F0 8B8700000000    mov eax, dword ptr [edi+00000000]
:004015F6 0FAFD8        imul ebx, eax
:004015F9 89D8          mov eax, ebx
:004015FB 89C3          mov ebx, eax
:004015FD 8B8704000000    mov eax, dword ptr [edi+00000004]
:00401603 39C3          cmp ebx, eax
:00401605 0F8420000000    je 0040162B

```

```

:0040160B 8B8700000000    mov eax, dword ptr [edi+00000000]
:00401611 89C3              mov ebx, eax
:00401613 B801000000      mov eax, 00000001
:00401618 29C3              sub ebx, eax
:0040161A 89D8              mov eax, ebx
:0040161C E8FFFBFFFF      call 00401220
:00401621 B871164000      mov eax, 00401671
:00401626 E8B5FBFFFF      call 004011E0

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401605(C)

```

|
:0040162B 8B8700000000    mov eax, dword ptr [edi+00000000]
:00401631 E8EAFBFFFF      call 00401220

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004015DD(C)

```

|
:00401636 B874164000      mov eax, 00401674
:0040163B E8A0FBFFFF      call 004011E0
:00401640 E8CBFBFFFF      call 00401210
:00401645 E93EFDFFFF      jmp 00401388
:0040164A 0000000000000000  BYTE 8 DUP(0)

```

```

:00401652 4E              dec esi
:00401653 3A20            cmp ah, byte ptr [eax]
:00401655 005241          add byte ptr [edx+41], dl
:00401658 49              dec ecx
:00401659 5A              pop edx
:0040165A 204355          and byte ptr [ebx+55], al
:0040165D 41              inc ecx
:0040165E 44              inc esp
:0040165F 52              push edx
:00401660 41              inc ecx
:00401661 44              inc esp
:00401662 41              inc ecx
:00401663 20444520        and byte ptr [ebp+2*eax+20], al
:00401667 003A            add byte ptr [edx], bh
:00401669 2000            and byte ptr [eax], al
:0040166B 45              inc ebp
:0040166C 52              push edx
:0040166D 52              push edx
:0040166E 4F              dec edi
:0040166F 52              push edx
:00401670 002E            add byte ptr [esi], ch

```

:00401672 2E

BYTE 02eh

:00401673 00000000000000000000

BYTE 10 DUP(0)

06

```

var IMPORTE, BILLETE, VUELTO, PESOS, CENTAVOS, VUELTOPEOS,
VUELTOCENTAVOS,
    IMPORTEOK, CENTAVOSOK, PESOSOK, BILLETEOK, VUELTOOK;

```

```

procedure ERROR;

```

```

begin

```

```

    writeln ('VALOR FUERA DE RANGO!')

```

```

end;

```

```

begin

```

```

    WRITELN ('*****');

```

```

    writeln ('VUELTO PARA IMPORTES PAGADOS CON UN BILLETE');

```

```

    WRITELN ('*****');

```

```

    writeln;

```

```

    IMPORTEOK := -1;

```

```

    while IMPORTEOK <> 0 do

```

```

        begin

```

```

            writeln ('IMPORTE (min $0.01 y max $100.00)');

```

```

            CENTAVOSOK := -1;

```

```

            while CENTAVOSOK <> 0 do

```

```

                begin

```

```

                    write ('CENTAVOS: '); readLn (CENTAVOS);

```

```

                    CENTAVOSOK := 0;

```

```

                    if CENTAVOS < 0 then CENTAVOSOK := -1;

```

```

                    if CENTAVOS > 99 then CENTAVOSOK := -1;

```

```

                    if CENTAVOSOK <> 0 then call ERROR

```

```

                end;

```

```

            PESOSOK := -1;

```

```

            while PESOSOK <> 0 do

```

```

                begin

```

```

                    write ('PESOS: '); readLn (PESOS);

```

```

                    PESOSOK := 0;

```

```

                    if PESOS < 0 then PESOSOK := -1;

```

```

                    if PESOS > 100 then PESOSOK := -1;

```

```

                    if PESOSOK <> 0 then call ERROR

```

```

                end;

```

```

            write ('IMPORTE: $', PESOS, '.');

```

```

            if CENTAVOS < 10 then write ('0');

```

```

            writeln (CENTAVOS);

```

```

            IMPORTE := PESOS * 100 + CENTAVOS;

```

```

            IMPORTEOK := 0;

```

```

            if IMPORTE < 1 then IMPORTEOK := -1;

```

```

            if IMPORTE > 10000 then IMPORTEOK := -1;

```

```

            if IMPORTEOK <> 0 then call ERROR

```

```

        end;

```

```

VUELTOOK := -1;
while VUELTOOK <> 0 do
begin
  BILLETEOK := -1;
  while BILLETEOK <> 0 do
  begin
    write ('BILLETE (min $2 y max $100): $'); readLn
(BILLETE);
    BILLETEOK := -1;
    if BILLETE = 2 then BILLETEOK := 0;
    if BILLETE = 5 then BILLETEOK := 0;
    if BILLETE = 10 then BILLETEOK := 0;
    if BILLETE = 20 then BILLETEOK := 0;
    if BILLETE = 50 then BILLETEOK := 0;
    if BILLETE = 100 then BILLETEOK := 0;
    if BILLETEOK <> 0 then writeln ('BILLETE INEXISTENTE!')
  end;
  VUELTO := BILLETE * 100 - IMPORTE;
  VUELTOOK := 0;
  if VUELTO < 0 then
  begin
    VUELTOOK := -1;
    writeln ('BILLETE INSUFICIENTE PARA EL PAGO!')
  end
end;

VUELTOPESOS := VUELTO / 100;
VUELTOCENTAVOS := VUELTO - VUELTOPESOS * 100;

write ('Su vuelto: $', VUELTOPESOS, '.');
if VUELTOCENTAVOS < 10 then write ('0');
writeln (VUELTOCENTAVOS);

writeln;

if VUELTO >= 5000 then
begin
  writeln ('1 billete de $50');
  VUELTO := VUELTO - 5000
end;

if VUELTO >= 2000 then
begin
  if VUELTO / 2000 = 1 then writeln ('1 billete de $20');
  if VUELTO / 2000 <> 1 then writeln ('2 billetes de $20');

```

```

    VUELTO := VUELTO - VUELTO / 2000 * 2000
end;

if VUELTO >= 1000 then
begin
    writeLn ('1 billete de $10');
    VUELTO := VUELTO - 1000
end;

if VUELTO >= 500 then
begin
    writeLn ('1 billete de $5');
    VUELTO := VUELTO - 500
end;

if VUELTO >= 200 then
begin
    if VUELTO / 200 = 1 then writeLn ('1 billete de $2');
    if VUELTO / 200 <> 1 then writeLn ('2 billetes de $2');
    VUELTO := VUELTO - VUELTO / 200 * 200
end;

if VUELTO >= 100 then
begin
    writeLn ('1 moneda de $1');
    VUELTO := VUELTO - 100
end;

if VUELTO >= 50 then
begin
    writeLn ('1 moneda de 50 centavos');
    VUELTO := VUELTO - 50
end;

if VUELTO >= 25 then
begin
    writeLn ('1 moneda de 25 centavos');
    VUELTO := VUELTO - 25
end;

if VUELTO >= 10 then
begin
    if VUELTO / 10 = 1 then writeLn ('1 moneda de 10 centavos');
    if VUELTO / 10 <> 1 then writeLn ('2 monedas de 10
centavos');
    VUELTO := VUELTO - VUELTO / 10 * 10

```

```
end;

if VUELTO >= 5 then
begin
  writeln ('1 moneda de 5 centavos');
  VUELTO := VUELTO - 5
end;

if VUELTO > 1 then writeln (VUELTO, ' monedas de 1 centavo');
if VUELTO = 1 then writeln ('1 moneda de 1 centavo')

end.
```



```

000003d0    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
000003e0    60 89 c6 30 c0 02 06 74    08 46 e8 e1 fe ff ff eb    `..0...t.F.....
000003f0    f2 61 90 c3 00 00 00 00    00 00 00 00 00 00 00 00    .a.....
00000400    04 30 e8 c9 fe ff ff c3    00 00 00 00 00 00 00 00    .0.....
00000410    b0 0d e8 b9 fe ff ff b0    0a e8 b2 fe ff ff c3 00    .....
00000420    3d 00 00 00 80 75 4e b0    2d e8 a2 fe ff ff b0 02    =....uN.-.....
00000430    e8 cb ff ff ff b0 01 e8    c4 ff ff ff b0 04 e8 bd    .....
00000440    ff ff ff b0 07 e8 b6 ff    ff ff b0 04 e8 af ff ff    .....
00000450    ff b0 08 e8 a8 ff ff ff    b0 03 e8 a1 ff ff ff b0    .....
00000460    06 e8 9a ff ff ff b0 04    e8 93 ff ff ff b0 08 e8    .....
00000470    8c ff ff ff c3 3d 00 00    00 00 7d 0b 50 b0 2d e8    .....=....}.P.-.
00000480    4c fe ff ff 58 f7 d8 3d    0a 00 00 00 0f 8c ef 00    L...X..=.....
00000490    00 00 3d 64 00 00 00 0f    8c d1 00 00 00 3d e8 03    ..=d.....=.
000004a0    00 00 0f 8c b3 00 00 00    3d 10 27 00 00 0f 8c 95    .....=.'.....
000004b0    00 00 00 3d a0 86 01 00    7c 7b 3d 40 42 0f 00 7c    ...=....|{@B..|
000004c0    61 3d 80 96 98 00 7c 47    3d 00 e1 f5 05 7c 2d 3d    a=....|G=....|-=
000004d0    00 ca 9a 3b 7c 13 ba 00    00 00 00 bb 00 ca 9a 3b    ...;|.....;
000004e0    f7 fb 52 e8 18 ff ff ff    58 ba 00 00 00 00 bb 00    ..R....X.....
000004f0    e1 f5 05 f7 fb 52 e8 05    ff ff ff 58 ba 00 00 00    .....R....X....
00000500    00 bb 80 96 98 00 f7 fb    52 e8 f2 fe ff ff 58 ba    .....R....X.
00000510    00 00 00 00 bb 40 42 0f    00 f7 fb 52 e8 df fe ff    .....@B....R....
00000520    ff 58 ba 00 00 00 00 bb    a0 86 01 00 f7 fb 52 e8    .X.....R.
00000530    cc fe ff ff 58 ba 00 00    00 00 bb 10 27 00 00 f7    ....X.....'...
00000540    fb 52 e8 b9 fe ff ff 58    ba 00 00 00 00 bb e8 03    .R....X.....
00000550    00 00 f7 fb 52 e8 a6 fe    ff ff 58 ba 00 00 00 00    ...R....X.....
00000560    bb 64 00 00 00 f7 fb 52    e8 93 fe ff ff 58 ba 00    .d....R....X..
00000570    00 00 00 bb 0a 00 00 00    f7 fb 52 e8 80 fe ff ff    .....R.....
00000580    58 e8 7a fe ff ff c3 00    ff 15 00 10 40 00 00 00    X.z.....@...
00000590    b9 00 00 00 00 b3 03 51    53 e8 a2 fd ff ff 5b 59    .....QS....[Y
000005a0    3c 0d 0f 84 34 01 00 00    3c 08 0f 84 94 00 00 00    <...4...<.....
000005b0    3c 2d 0f 84 09 01 00 00    3c 30 7c db 3c 39 7f d7    <-.....<0|.<9..
000005c0    2c 30 80 fb 00 74 d0 80    fb 02 75 0c 81 f9 00 00    ,0...t....u.....
000005d0    00 00 75 04 3c 00 74 bf    80 fb 03 75 0a 3c 00 75    ..u.<.t....u.<.u
000005e0    04 b3 00 eb 02 b3 01 81    f9 cc cc cc 0c 7f a8 81    .....
000005f0    f9 34 33 33 f3 7c a0 88    c7 b8 0a 00 00 00 f7 e9    .433.|.....
00000600    3d 08 00 00 80 74 11 3d    f8 ff ff 7f 75 13 80 ff    =....t.=...u...
00000610    07 7e 0e e9 7f ff ff ff    80 ff 08 0f 8f 76 ff ff    .~.....v..
00000620    ff b9 00 00 00 00 88 f9    80 fb 02 74 04 01 c1 eb    .....t....
00000630    03 29 c8 91 88 f8 51 53    e8 c3 fd ff ff 5b 59 e9    .)....QS....[Y.
00000640    53 ff ff ff 80 fb 03 0f    84 4a ff ff ff 51 53 b0    S.....J....QS.
00000650    08 e8 7a fc ff ff b0 20    e8 73 fc ff ff b0 08 e8    .z....s.....
00000660    6c fc ff ff 5b 59 80 fb    00 75 07 b3 03 e9 25 ff    l...[Y...u...%.
00000670    ff ff 80 fb 02 75 0f 81    f9 00 00 00 00 75 07 b3    .....u.....u..
00000680    03 e9 11 ff ff ff 89 c8    b9 0a 00 00 00 ba 00 00    .....
00000690    00 00 3d 00 00 00 00 7d    08 f7 d8 f7 f9 f7 d8 eb    ..=....}.....
000006a0    02 f7 f9 89 c1 81 f9 00    00 00 00 0f 85 e6 fe ff    .....
000006b0    ff 80 fb 02 0f 84 dd fe    ff ff b3 03 e9 d6 fe ff    .....
000006c0    ff 80 fb 03 0f 85 cd fe    ff ff b0 2d 51 53 e8 fd    .....-QS..
000006d0    fb ff ff 5b 59 b3 02 e9    bb fe ff ff 80 fb 03 0f    ...[Y.....
000006e0    84 b2 fe ff ff 80 fb 02    75 0c 81 f9 00 00 00 00    .....u.....
000006f0    0f 84 a1 fe ff ff 51 e8    14 fd ff ff 59 89 c8 c3    .....Q....Y...
00000700    bf f2 1d 40 00 e9 10 00    00 00 b8 22 1e 40 00 e8    ...@.....".@..
00000710    cc fc ff ff e8 f7 fc ff    ff c3 b8 38 1e 40 00 e8    .....8.@..
00000720    bc fc ff ff e8 e7 fc ff    ff b8 64 1e 40 00 e8 ad    .....d.@...
00000730    fc ff ff e8 d8 fc ff ff    b8 90 1e 40 00 e8 9e fc    .....@.....
00000740    ff ff e8 c9 fc ff ff b8    bc 1e 40 00 e8 8f fc ff    .....@.....
00000750    ff e8 ba fc ff ff b8 01    00 00 00 f7 d8 89 87 1c    .....
00000760    00 00 00 8b 87 1c 00 00    00 89 c3 b8 00 00 00 00    .....
00000770    39 c3 0f 84 39 02 00 00    b8 bd 1e 40 00 e8 5e fc    9...9.....@..^.
00000780    ff ff e8 89 fc ff ff b8    01 00 00 00 f7 d8 89 87    .....
00000790    20 00 00 00 8b 87 20 00    00 00 89 c3 b8 00 00 00    .....
000007a0    00 39 c3 0f 84 83 00 00    00 b8 df 1e 40 00 e8 2d    .9.....@..-

```

000007b0	fc ff ff e8 d8 fd ff ff	89 87 10 00 00 00 b8 00
000007c0	00 00 00 89 87 20 00 00	00 8b 87 10 00 00 00 89
000007d0	c3 b8 00 00 00 00 39 c3	0f 8d 0d 00 00 00 b8 019.....
000007e0	00 00 00 f7 d8 89 87 20	00 00 00 8b 87 10 00 00
000007f0	00 89 c3 b8 63 00 00 00	39 c3 0f 8e 0d 00 00 00c...9.....
00000800	b8 01 00 00 00 f7 d8 89	87 20 00 00 00 8b 87 20
00000810	00 00 00 89 c3 b8 00 00	00 00 39 c3 0f 84 05 009.....
00000820	00 00 e8 e3 fe ff ff e9	68 ff ff ff b8 01 00 00h.....
00000830	00 f7 d8 89 87 24 00 00	00 8b 87 24 00 00 00 89\$......\$....
00000840	c3 b8 00 00 00 00 39 c3	0f 84 83 00 00 00 b8 ea9.....
00000850	1e 40 00 e8 88 fb ff ff	e8 33 fd ff ff 89 87 0c	.@.....3.....
00000860	00 00 00 b8 00 00 00 00	89 87 24 00 00 00 8b 87\$......
00000870	0c 00 00 00 89 c3 b8 00	00 00 00 39 c3 0f 8d 0d9.....
00000880	00 00 00 b8 01 00 00 00	f7 d8 89 87 24 00 00 00\$....
00000890	8b 87 0c 00 00 00 89 c3	b8 64 00 00 00 39 c3 0fd...9..
000008a0	8e 0d 00 00 00 b8 01 00	00 00 f7 d8 89 87 24 00\$....
000008b0	00 00 8b 87 24 00 00 00	89 c3 b8 00 00 00 00 39\$......9
000008c0	c3 0f 84 05 00 00 00 e8	3e fe ff ff e9 68 ff ff>...h..
000008d0	ff b8 f2 1e 40 00 e8 05	fb ff ff 8b 87 0c 00 00@.....
000008e0	00 e8 3a fb ff ff b8 fd	1e 40 00 e8 f0 fa ff ff	..:.....@.....
000008f0	8b 87 10 00 00 00 89 c3	b8 0a 00 00 00 39 c3 0f9..
00000900	8d 0a 00 00 00 b8 ff 1e	40 00 e8 d1 fa ff ff 8b@.....
00000910	87 10 00 00 00 e8 06 fb	ff ff e8 f1 fa ff ff 8b
00000920	87 0c 00 00 00 89 c3 b8	64 00 00 00 0f af d8 89d.....
00000930	d8 89 c3 8b 87 10 00 00	00 01 c3 89 d8 89 87 00
00000940	00 00 00 b8 00 00 00 00	89 87 1c 00 00 00 8b 87
00000950	00 00 00 00 89 c3 b8 01	00 00 00 39 c3 0f 8d 0d9.....
00000960	00 00 00 b8 01 00 00 00	f7 d8 89 87 1c 00 00 00
00000970	8b 87 00 00 00 00 89 c3	b8 10 27 00 00 39 c3 0f'.9..
00000980	8e 0d 00 00 00 b8 01 00	00 00 f7 d8 89 87 1c 00
00000990	00 00 8b 87 1c 00 00 00	89 c3 b8 00 00 00 00 399
000009a0	c3 0f 84 05 00 00 00 e8	5e fd ff ff e9 b2 fd ff^.....
000009b0	ff b8 01 00 00 00 f7 d8	89 87 2c 00 00 00 8b 87,
000009c0	2c 00 00 00 89 c3 b8 00	00 00 00 39 c3 0f 84 92	,.....9.....
000009d0	01 00 00 b8 01 00 00 00	f7 d8 89 87 28 00 00 00(...
000009e0	8b 87 28 00 00 00 89 c3	b8 00 00 00 00 39 c3 0f	..(.....9..
000009f0	84 0b 01 00 00 b8 01 1f	40 00 e8 e1 f9 ff ff e8@.....
00000a00	8c fb ff ff 89 87 04 00	00 00 b8 01 00 00 00 f7
00000a10	d8 89 87 28 00 00 00 8b	87 04 00 00 00 89 c3 b8	...(.....
00000a20	02 00 00 00 39 c3 0f 85	0b 00 00 00 b8 00 00 00	...9.....
00000a30	00 89 87 28 00 00 00 8b	87 04 00 00 00 89 c3 b8	...(.....
00000a40	05 00 00 00 39 c3 0f 85	0b 00 00 00 b8 00 00 00	...9.....
00000a50	00 89 87 28 00 00 00 8b	87 04 00 00 00 89 c3 b8	...(.....
00000a60	0a 00 00 00 39 c3 0f 85	0b 00 00 00 b8 00 00 00	...9.....
00000a70	00 89 87 28 00 00 00 8b	87 04 00 00 00 89 c3 b8	...(.....
00000a80	14 00 00 00 39 c3 0f 85	0b 00 00 00 b8 00 00 00	...9.....
00000a90	00 89 87 28 00 00 00 8b	87 04 00 00 00 89 c3 b8	...(.....
00000aa0	32 00 00 00 39 c3 0f 85	0b 00 00 00 b8 00 00 00	2...9.....
00000ab0	00 89 87 28 00 00 00 8b	87 04 00 00 00 89 c3 b8	...(.....
00000ac0	64 00 00 00 39 c3 0f 85	0b 00 00 00 b8 00 00 00	d...9.....
00000ad0	00 89 87 28 00 00 00 8b	87 28 00 00 00 89 c3 b8	...(.....(.....
00000ae0	00 00 00 00 39 c3 0f 84	0f 00 00 00 b8 20 1f 409..... @
00000af0	00 e8 ea f8 ff ff e8 15	f9 ff ff e9 e0 fe ff ff
00000b00	8b 87 04 00 00 00 89 c3	b8 64 00 00 00 0f af d8d.....
00000b10	89 d8 89 c3 8b 87 00 00	00 00 29 c3 89 d8 89 87),.....
00000b20	08 00 00 00 b8 00 00 00	00 89 87 2c 00 00 00 8b,
00000b30	87 08 00 00 00 89 c3 b8	00 00 00 00 39 c3 0f 8d9...
00000b40	1c 00 00 00 b8 01 00 00	00 f7 d8 89 87 2c 00 00,
00000b50	00 b8 35 1f 40 00 e8 85	f8 ff ff e8 b0 f8 ff ff	..5.@.....
00000b60	e9 59 fe ff ff 8b 87 08	00 00 00 89 c3 b8 64 00	.Y.....d.
00000b70	00 00 93 99 f7 fb 93 89	d8 89 87 14 00 00 00 8b
00000b80	87 08 00 00 00 89 c3 53	8b 87 14 00 00 00 89 c3S.....

00000b90	b8 64 00 00 00 0f af d8	89 d8 5b 29 c3 89 d8 89	.d.....[)....
00000ba0	87 18 00 00 00 b8 58 1f	40 00 e8 31 f8 ff ff 8bX.@..1....
00000bb0	87 14 00 00 00 e8 66 f8	ff ff b8 65 1f 40 00 e8f....e.@..
00000bc0	1c f8 ff ff 8b 87 18 00	00 00 89 c3 b8 0a 00 00
00000bd0	00 39 c3 0f 8d 0a 00 00	00 b8 67 1f 40 00 e8 fd	.9.....g.@...
00000be0	f7 ff ff 8b 87 18 00 00	00 e8 32 f8 ff ff e8 1d2.....
00000bf0	f8 ff ff b8 69 1f 40 00	e8 e3 f7 ff ff e8 0e f8i.@.....
00000c00	ff ff 8b 87 08 00 00 00	89 c3 b8 88 13 00 00 399
00000c10	c3 0f 8c 26 00 00 00 b8	6a 1f 40 00 e8 bf f7 ff	...&....j.@.....
00000c20	ff e8 ea f7 ff ff 8b 87	08 00 00 00 89 c3 b8 88
00000c30	13 00 00 29 c3 89 d8 89	87 08 00 00 00 8b 87 08	...).
00000c40	00 00 00 89 c3 b8 d0 07	00 00 39 c3 0f 8c 94 009.....
00000c50	00 00 8b 87 08 00 00 00	89 c3 b8 d0 07 00 00 93
00000c60	99 f7 fb 93 89 d8 89 c3	b8 01 00 00 00 39 c3 0f9..
00000c70	85 0f 00 00 00 b8 7b 1f	40 00 e8 61 f7 ff ff e8{.@..a....
00000c80	8c f7 ff ff 8b 87 08 00	00 00 89 c3 b8 d0 07 00
00000c90	00 93 99 f7 fb 93 89 d8	89 c3 b8 01 00 00 00 399
00000ca0	c3 0f 84 0f 00 00 00 b8	8c 1f 40 00 e8 2f f7 ff@../..
00000cb0	ff e8 5a f7 ff ff 8b 87	08 00 00 00 89 c3 53 8b	..Z.....S.
00000cc0	87 08 00 00 00 89 c3 b8	d0 07 00 00 93 99 f7 fb
00000cd0	93 b8 d0 07 00 00 0f af	d8 89 d8 5b 29 c3 89 d8[)...
00000ce0	89 87 08 00 00 00 8b 87	08 00 00 00 89 c3 b8 e8
00000cf0	03 00 00 39 c3 0f 8c 26	00 00 00 b8 9e 1f 40 00	...9...&.....@.
00000d00	e8 db f6 ff ff e8 06 f7	ff ff 8b 87 08 00 00 00
00000d10	89 c3 b8 e8 03 00 00 29	c3 89 d8 89 87 08 00 00)
00000d20	00 8b 87 08 00 00 00 89	c3 b8 f4 01 00 00 39 c39.
00000d30	0f 8c 26 00 00 00 b8 af	1f 40 00 e8 a0 f6 ff ff	..&.....@.....
00000d40	e8 cb f6 ff ff 8b 87 08	00 00 00 89 c3 b8 f4 01
00000d50	00 00 29 c3 89 d8 89 87	08 00 00 00 8b 87 08 00	..).
00000d60	00 00 89 c3 b8 c8 00 00	00 39 c3 0f 8c 94 00 009.....
00000d70	00 8b 87 08 00 00 00 89	c3 b8 c8 00 00 00 93 99
00000d80	f7 fb 93 89 d8 89 c3 b8	01 00 00 00 39 c3 0f 859...
00000d90	0f 00 00 00 b8 bf 1f 40	00 e8 42 f6 ff ff e8 6d@..B....m
00000da0	f6 ff ff 8b 87 08 00 00	00 89 c3 b8 c8 00 00 00
00000db0	93 99 f7 fb 93 89 d8 89	c3 b8 01 00 00 00 39 c39.
00000dc0	0f 84 0f 00 00 00 b8 cf	1f 40 00 e8 10 f6 ff ff@.....
00000dd0	e8 3b f6 ff ff 8b 87 08	00 00 00 89 c3 53 8b 87	./.....S..
00000de0	08 00 00 00 89 c3 b8 c8	00 00 00 93 99 f7 fb 93
00000df0	b8 c8 00 00 00 0f af d8	89 d8 5b 29 c3 89 d8 89[)....
00000e00	00 00 00 00 8b 87 08	00 00 00 89 c3 b8 64 00d.
00000e10	00 00 39 c3 0f 8c 26 00	00 00 b8 e0 1f 40 00 e8	..9...&.....@..
00000e20	bc f5 ff ff e8 e7 f5 ff	ff 8b 87 08 00 00 00 89
00000e30	c3 b8 64 00 00 00 29 c3	89 d8 89 87 08 00 00 00	..d...)
00000e40	8b 87 08 00 00 00 89 c3	b8 32 00 00 00 39 c3 0f2...9..
00000e50	8c 26 00 00 00 b8 ef 1f	40 00 e8 81 f5 ff ff e8	..&.....@.....
00000e60	ac f5 ff ff 8b 87 08 00	00 00 89 c3 b8 32 00 002..
00000e70	00 29 c3 89 d8 89 87 08	00 00 00 8b 87 08 00 00	.)
00000e80	00 89 c3 b8 19 00 00 00	39 c3 0f 8c 26 00 00 009...&...
00000e90	b8 07 20 40 00 e8 46 f5	ff ff e8 71 f5 ff ff 8b	.. @..F....q....
00000ea0	87 08 00 00 00 89 c3 b8	19 00 00 00 29 c3 89 d8)
00000eb0	89 87 08 00 00 00 8b 87	08 00 00 00 89 c3 b8 0a
00000ec0	00 00 00 39 c3 0f 8c 94	00 00 00 8b 87 08 00 00	...9.....
00000ed0	00 89 c3 b8 0a 00 00 00	93 99 f7 fb 93 89 d8 89
00000ee0	c3 b8 01 00 00 00 39 c3	0f 85 0f 00 00 00 b8 1f9.....
00000ef0	20 40 00 e8 e8 f4 ff ff	e8 13 f5 ff ff 8b 87 08	@.....
00000f00	00 00 00 89 c3 b8 0a 00	00 00 93 99 f7 fb 93 89
00000f10	d8 89 c3 b8 01 00 00 00	39 c3 0f 84 0f 00 00 009.....
00000f20	b8 37 20 40 00 e8 b6 f4	ff ff e8 e1 f4 ff ff 8b	.7 @.....
00000f30	87 08 00 00 00 89 c3 53	8b 87 08 00 00 00 89 c3S.....
00000f40	b8 0a 00 00 00 93 99 f7	fb 93 b8 0a 00 00 00 0f
00000f50	af d8 89 d8 5b 29 c3 89	d8 89 87 08 00 00 00 8b[)
00000f60	87 08 00 00 00 89 c3 b8	05 00 00 00 39 c3 0f 8c9...

00000f70	26 00 00 00 b8 50 20 40	00 e8 62 f4 ff ff e8 8d	&....P @..b.....
00000f80	f4 ff ff 8b 87 08 00 00	00 89 c3 b8 05 00 00 00
00000f90	29 c3 89 d8 89 87 08 00	00 00 8b 87 08 00 00 00).....
00000fa0	89 c3 b8 01 00 00 00 39	c3 0f 8e 1a 00 00 00 8b9.....
00000fb0	87 08 00 00 00 e8 66 f4	ff ff b8 67 20 40 00 e8f....g @..
00000fc0	1c f4 ff ff e8 47 f4 ff	ff 8b 87 08 00 00 00 89G.....
00000fd0	c3 b8 01 00 00 00 39 c3	0f 85 0f 00 00 00 b8 7d9.....}
00000fe0	20 40 00 e8 f8 f3 ff ff	e8 23 f4 ff ff e9 96 f5	@.....#.....
00000ff0	ff ff 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00001000	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00001010	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00001020	00 00 56 41 4c 4f 52 20	46 55 45 52 41 20 44 45	..VALOR FUERA DE
00001030	20 52 41 4e 47 4f 21 00	2a 2a 2a 2a 2a 2a 2a 2a	RANGO!.*****
00001040	2a 2a 2a 2a 2a 2a 2a 2a	2a 2a 2a 2a 2a 2a 2a 2a	*****
00001050	2a 2a 2a 2a 2a 2a 2a 2a	2a 2a 2a 2a 2a 2a 2a 2a	*****
00001060	2a 2a 2a 00 56 55 45 4c	54 4f 20 50 41 52 41 20	***.VUELTO PARA
00001070	49 4d 50 4f 52 54 45 53	20 50 41 47 41 44 4f 53	IMPORTES PAGADOS
00001080	20 43 4f 4e 20 55 4e 20	42 49 4c 4c 45 54 45 00	CON UN BILLETE.
00001090	2a 2a 2a 2a 2a 2a 2a 2a	2a 2a 2a 2a 2a 2a 2a 2a	*****
000010a0	2a 2a 2a 2a 2a 2a 2a 2a	2a 2a 2a 2a 2a 2a 2a 2a	*****
000010b0	2a 2a 2a 2a 2a 2a 2a 2a	2a 2a 2a 00 00 49 4d 50	*****..IMP
000010c0	4f 52 54 45 20 28 6d 69	6e 20 24 30 2e 30 31 20	ORTE (min \$0.01
000010d0	79 20 6d 61 78 20 24 31	30 30 2e 30 30 29 00 43	y max \$100.00).C
000010e0	45 4e 54 41 56 4f 53 3a	20 00 50 45 53 4f 53 3a	ENTAVOS: .PESOS:
000010f0	20 00 49 4d 50 4f 52 54	45 3a 20 24 00 2e 00 30	.IMPORTE: \$...0
00001100	00 42 49 4c 4c 45 54 45	20 28 6d 69 6e 20 24 32	.BILLETE (min \$2
00001110	20 79 20 6d 61 78 20 24	31 30 30 29 3a 20 24 00	y max \$100): \$.
00001120	42 49 4c 4c 45 54 45 20	49 4e 45 58 49 53 54 45	BILLETE INEXISTE
00001130	4e 54 45 21 00 42 49 4c	4c 45 54 45 20 49 4e 53	NTE!.BILLETE INS
00001140	55 46 49 43 49 45 4e 54	45 20 50 41 52 41 20 45	UFICIENTE PARA E
00001150	4c 20 50 41 47 4f 21 00	53 75 20 76 75 65 6c 74	L PAGO!.Su vuelto
00001160	6f 3a 20 24 00 2e 00 30	00 00 31 20 62 69 6c 6c	o: \$...0..1 bill
00001170	65 74 65 20 64 65 20 24	35 30 00 31 20 62 69 6c	ete de \$50.1 bil
00001180	6c 65 74 65 20 64 65 20	24 32 30 00 32 20 62 69	lete de \$20.2 bi
00001190	6c 6c 65 74 65 73 20 64	65 20 24 32 30 00 31 20	lletes de \$20.1
000011a0	62 69 6c 6c 65 74 65 20	64 65 20 24 31 30 00 31	billete de \$10.1
000011b0	20 62 69 6c 6c 65 74 65	20 64 65 20 24 35 00 31	billete de \$5.1
000011c0	20 62 69 6c 6c 65 74 65	20 64 65 20 24 32 00 32	billete de \$2.2
000011d0	20 62 69 6c 6c 65 74 65	73 20 64 65 20 24 32 00	billetes de \$2.
000011e0	31 20 6d 6f 6e 65 64 61	20 64 65 20 24 31 00 31	1 moneda de \$1.1
000011f0	20 6d 6f 6e 65 64 61 20	64 65 20 35 30 20 63 65	moneda de 50 ce
00001200	6e 74 61 76 6f 73 00 31	20 6d 6f 6e 65 64 61 20	ntavos.1 moneda
00001210	64 65 20 32 35 20 63 65	6e 74 61 76 6f 73 00 31	de 25 centavos.1
00001220	20 6d 6f 6e 65 64 61 20	64 65 20 31 30 20 63 65	moneda de 10 ce
00001230	6e 74 61 76 6f 73 00 32	20 6d 6f 6e 65 64 61 73	ntavos.2 monedas
00001240	20 64 65 20 31 30 20 63	65 6e 74 61 76 6f 73 00	de 10 centavos.
00001250	31 20 6d 6f 6e 65 64 61	20 64 65 20 35 20 63 65	1 moneda de 5 ce
00001260	6e 74 61 76 6f 73 00 20	6d 6f 6e 65 64 61 73 20	ntavos. monedas
00001270	64 65 20 31 20 63 65 6e	74 61 76 6f 00 31 20 6d	de 1 centavo.1 m
00001280	6f 6e 65 64 61 20 64 65	20 31 20 63 65 6e 74 61	oneda de 1 centa
00001290	76 6f 00 00 00 00 00 00	00 00 00 00 00 00 00 00	vo.....
000012a0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000012b0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000012c0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000012d0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000012e0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000012f0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

desensamblado

```

:00401500 BFF21D4000      mov edi, 00401DF2
:00401505 E910000000      jmp 0040151A
:0040150A B8221E4000      mov eax, 00401E22
:0040150F E8CCFCFFFF      call 004011E0
:00401514 E8F7FCFFFF      call 00401210
:00401519 C3              ret

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401505(U)

```

|
:0040151A B8381E4000      mov eax, 00401E38
:0040151F E8BCFCFFFF      call 004011E0
:00401524 E8E7FCFFFF      call 00401210
:00401529 B8641E4000      mov eax, 00401E64
:0040152E E8ADFCFFFF      call 004011E0
:00401533 E8D8FCFFFF      call 00401210
:00401538 B8901E4000      mov eax, 00401E90
:0040153D E89EFCFFFF      call 004011E0
:00401542 E8C9FCFFFF      call 00401210
:00401547 B8BC1E4000      mov eax, 00401EBC
:0040154C E88FFCFFFF      call 004011E0
:00401551 E8BAFCFFFF      call 00401210
:00401556 B801000000      mov eax, 00000001
:0040155B F7D8          neg eax
:0040155D 89871C000000      mov dword ptr [edi+0000001C], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004017AC(U)

```

|
:00401563 8B871C000000      mov eax, dword ptr [edi+0000001C]
:00401569 89C3              mov ebx, eax
:0040156B B800000000      mov eax, 00000000
:00401570 39C3              cmp ebx, eax
:00401572 0F8439020000      je 004017B1
:00401578 B8BD1E4000      mov eax, 00401EBD
:0040157D E85EFCFFFF      call 004011E0
:00401582 E889FCFFFF      call 00401210
:00401587 B801000000      mov eax, 00000001
:0040158C F7D8          neg eax
:0040158E 898720000000      mov dword ptr [edi+00000020], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401627(U)

|

```
:00401594 8B8720000000    mov eax, dword ptr [edi+00000020]
:0040159A 89C3              mov ebx, eax
:0040159C B800000000       mov eax, 00000000
:004015A1 39C3              cmp ebx, eax
:004015A3 0F8483000000     je 0040162C
:004015A9 B8DF1E4000       mov eax, 00401EDF
:004015AE E82DFCFFFF       call 004011E0
:004015B3 E8D8FDFFFF       call 00401390
:004015B8 898710000000     mov dword ptr [edi+00000010], eax
:004015BE B800000000       mov eax, 00000000
:004015C3 898720000000     mov dword ptr [edi+00000020], eax
:004015C9 8B8710000000     mov eax, dword ptr [edi+00000010]
:004015CF 89C3              mov ebx, eax
:004015D1 B800000000       mov eax, 00000000
:004015D6 39C3              cmp ebx, eax
:004015D8 0F8D0D000000     jnl 004015EB
:004015DE B801000000       mov eax, 00000001
:004015E3 F7D8              neg eax
:004015E5 898720000000     mov dword ptr [edi+00000020], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:004015D8(C)

|

```
:004015EB 8B8710000000     mov eax, dword ptr [edi+00000010]
:004015F1 89C3              mov ebx, eax
:004015F3 B863000000       mov eax, 00000063
:004015F8 39C3              cmp ebx, eax
:004015FA 0F8E0D000000     jle 0040160D
:00401600 B801000000       mov eax, 00000001
:00401605 F7D8              neg eax
:00401607 898720000000     mov dword ptr [edi+00000020], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:004015FA(C)

|

```
:0040160D 8B8720000000     mov eax, dword ptr [edi+00000020]
:00401613 89C3              mov ebx, eax
:00401615 B800000000       mov eax, 00000000
:0040161A 39C3              cmp ebx, eax
:0040161C 0F8405000000     je 00401627
:00401622 E8E3FEFFFF       call 0040150A
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:0040161C(C)

```
|
:00401627 E968FFFFFF          jmp 00401594
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004015A3(C)

```
|
:0040162C B801000000          mov eax, 00000001
:00401631 F7D8              neg eax
:00401633 898724000000       mov dword ptr [edi+00000024], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004016CC(U)

```
|
:00401639 8B8724000000       mov eax, dword ptr [edi+00000024]
:0040163F 89C3              mov ebx, eax
:00401641 B800000000       mov eax, 00000000
:00401646 39C3              cmp ebx, eax
:00401648 0F8483000000       je 004016D1
:0040164E B8EA1E4000       mov eax, 00401EEA
:00401653 E888FBFFFF       call 004011E0
:00401658 E833FDFFFF       call 00401390
:0040165D 89870C000000       mov dword ptr [edi+0000000C], eax
:00401663 B800000000       mov eax, 00000000
:00401668 898724000000       mov dword ptr [edi+00000024], eax
:0040166E 8B870C000000       mov eax, dword ptr [edi+0000000C]
:00401674 89C3              mov ebx, eax
:00401676 B800000000       mov eax, 00000000
:0040167B 39C3              cmp ebx, eax
:0040167D 0F8D0D000000       jnl 00401690
:00401683 B801000000       mov eax, 00000001
:00401688 F7D8              neg eax
:0040168A 898724000000       mov dword ptr [edi+00000024], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040167D(C)

```
|
:00401690 8B870C000000       mov eax, dword ptr [edi+0000000C]
:00401696 89C3              mov ebx, eax
:00401698 B864000000       mov eax, 00000064
:0040169D 39C3              cmp ebx, eax
:0040169F 0F8E0D000000       jle 004016B2
:004016A5 B801000000       mov eax, 00000001
:004016AA F7D8              neg eax
:004016AC 898724000000       mov dword ptr [edi+00000024], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

```

|:0040169F(C)
|
:004016B2 8B8724000000      mov eax, dword ptr [edi+00000024]
:004016B8 89C3              mov ebx, eax
:004016BA B800000000      mov eax, 00000000
:004016BF 39C3              cmp ebx, eax
:004016C1 0F8405000000      je 004016CC
:004016C7 E83EFEFFFF      call 0040150A

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004016C1(C)

```

|
:004016CC E968FFFFFF      jmp 00401639

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401648(C)

```

|
:004016D1 B8F21E4000      mov eax, 00401EF2
:004016D6 E805FBFFFF      call 004011E0
:004016DB 8B870C000000      mov eax, dword ptr [edi+0000000C]
:004016E1 E83AFBFFFF      call 00401220
:004016E6 B8FD1E4000      mov eax, 00401EFD
:004016EB E8F0FAFFFF      call 004011E0
:004016F0 8B8710000000      mov eax, dword ptr [edi+00000010]
:004016F6 89C3              mov ebx, eax
:004016F8 B80A000000      mov eax, 0000000A
:004016FD 39C3              cmp ebx, eax
:004016FF 0F8D0A000000      jnl 0040170F
:00401705 B8FF1E4000      mov eax, 00401EFF
:0040170A E8D1FAFFFF      call 004011E0

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004016FF(C)

```

|
:0040170F 8B8710000000      mov eax, dword ptr [edi+00000010]
:00401715 E806FBFFFF      call 00401220
:0040171A E8F1FAFFFF      call 00401210
:0040171F 8B870C000000      mov eax, dword ptr [edi+0000000C]
:00401725 89C3              mov ebx, eax
:00401727 B864000000      mov eax, 00000064
:0040172C 0FAFD8          imul ebx, eax
:0040172F 89D8          mov eax, ebx
:00401731 89C3          mov ebx, eax
:00401733 8B8710000000      mov eax, dword ptr [edi+00000010]
:00401739 01C3          add ebx, eax
:0040173B 89D8          mov eax, ebx

```

```

:0040173D 898700000000    mov dword ptr [edi+00000000], eax
:00401743 B800000000    mov eax, 00000000
:00401748 89871C000000    mov dword ptr [edi+0000001C], eax
:0040174E 8B8700000000    mov eax, dword ptr [edi+00000000]
:00401754 89C3    mov ebx, eax
:00401756 B801000000    mov eax, 00000001
:0040175B 39C3    cmp ebx, eax
:0040175D 0F8D0D000000    jnl 00401770
:00401763 B801000000    mov eax, 00000001
:00401768 F7D8    neg eax
:0040176A 89871C000000    mov dword ptr [edi+0000001C], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040175D(C)

```

|
:00401770 8B8700000000    mov eax, dword ptr [edi+00000000]
:00401776 89C3    mov ebx, eax
:00401778 B810270000    mov eax, 00002710
:0040177D 39C3    cmp ebx, eax
:0040177F 0F8E0D000000    jle 00401792
:00401785 B801000000    mov eax, 00000001
:0040178A F7D8    neg eax
:0040178C 89871C000000    mov dword ptr [edi+0000001C], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040177F(C)

```

|
:00401792 8B871C000000    mov eax, dword ptr [edi+0000001C]
:00401798 89C3    mov ebx, eax
:0040179A B800000000    mov eax, 00000000
:0040179F 39C3    cmp ebx, eax
:004017A1 0F8405000000    je 004017AC
:004017A7 E85EFDFFFF    call 0040150A

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004017A1(C)

```

|
:004017AC E9B2FDFFFF    jmp 00401563

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401572(C)

```

|
:004017B1 B801000000    mov eax, 00000001
:004017B6 F7D8    neg eax
:004017B8 89872C000000    mov dword ptr [edi+0000002C], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401960(U)

```
|
:004017BE 8B872C000000    mov eax, dword ptr [edi+0000002C]
:004017C4 89C3                mov ebx, eax
:004017C6 B800000000          mov eax, 00000000
:004017CB 39C3                cmp ebx, eax
:004017CD 0F8492010000        je 00401965
:004017D3 B801000000          mov eax, 00000001
:004017D8 F7D8                neg eax
:004017DA 898728000000        mov dword ptr [edi+00000028], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004018FB(U)

```
|
:004017E0 8B8728000000        mov eax, dword ptr [edi+00000028]
:004017E6 89C3                mov ebx, eax
:004017E8 B800000000          mov eax, 00000000
:004017ED 39C3                cmp ebx, eax
:004017EF 0F840B010000        je 00401900
:004017F5 B8011F4000          mov eax, 00401F01
:004017FA E8E1F9FFFF          call 004011E0
:004017FF E88CFBFFFF          call 00401390
:00401804 898704000000        mov dword ptr [edi+00000004], eax
:0040180A B801000000          mov eax, 00000001
:0040180F F7D8                neg eax
:00401811 898728000000        mov dword ptr [edi+00000028], eax
:00401817 8B8704000000        mov eax, dword ptr [edi+00000004]
:0040181D 89C3                mov ebx, eax
:0040181F B802000000          mov eax, 00000002
:00401824 39C3                cmp ebx, eax
:00401826 0F850B000000        jne 00401837
:0040182C B800000000          mov eax, 00000000
:00401831 898728000000        mov dword ptr [edi+00000028], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401826(C)

```
|
:00401837 8B8704000000        mov eax, dword ptr [edi+00000004]
:0040183D 89C3                mov ebx, eax
:0040183F B805000000          mov eax, 00000005
:00401844 39C3                cmp ebx, eax
:00401846 0F850B000000        jne 00401857
:0040184C B800000000          mov eax, 00000000
:00401851 898728000000        mov dword ptr [edi+00000028], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401846(C)

```
|
:00401857 8B8704000000    mov eax, dword ptr [edi+00000004]
:0040185D 89C3                mov ebx, eax
:0040185F B80A000000    mov eax, 0000000A
:00401864 39C3                cmp ebx, eax
:00401866 0F850B000000    jne 00401877
:0040186C B800000000    mov eax, 00000000
:00401871 898728000000    mov dword ptr [edi+00000028], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401866(C)

```
|
:00401877 8B8704000000    mov eax, dword ptr [edi+00000004]
:0040187D 89C3                mov ebx, eax
:0040187F B814000000    mov eax, 00000014
:00401884 39C3                cmp ebx, eax
:00401886 0F850B000000    jne 00401897
:0040188C B800000000    mov eax, 00000000
:00401891 898728000000    mov dword ptr [edi+00000028], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401886(C)

```
|
:00401897 8B8704000000    mov eax, dword ptr [edi+00000004]
:0040189D 89C3                mov ebx, eax
:0040189F B832000000    mov eax, 00000032
:004018A4 39C3                cmp ebx, eax
:004018A6 0F850B000000    jne 004018B7
:004018AC B800000000    mov eax, 00000000
:004018B1 898728000000    mov dword ptr [edi+00000028], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004018A6(C)

```
|
:004018B7 8B8704000000    mov eax, dword ptr [edi+00000004]
:004018BD 89C3                mov ebx, eax
:004018BF B864000000    mov eax, 00000064
:004018C4 39C3                cmp ebx, eax
:004018C6 0F850B000000    jne 004018D7
:004018CC B800000000    mov eax, 00000000
:004018D1 898728000000    mov dword ptr [edi+00000028], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004018C6(C)


```

|
:004018D7 8B8728000000    mov eax, dword ptr [edi+00000028]
:004018DD 89C3                 mov ebx, eax
:004018DF B800000000          mov eax, 00000000
:004018E4 39C3                 cmp ebx, eax
:004018E6 0F840F000000        je 004018FB
:004018EC B8201F4000          mov eax, 00401F20
:004018F1 E8EAF8FFFF          call 004011E0
:004018F6 E815F9FFFF          call 00401210

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004018E6(C)

```

|
:004018FB E9E0FEFFFF          jmp 004017E0

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004017EF(C)

```

|
:00401900 8B8704000000    mov eax, dword ptr [edi+00000004]
:00401906 89C3                 mov ebx, eax
:00401908 B864000000          mov eax, 00000064
:0040190D 0FAFD8            imul ebx, eax
:00401910 89D8                 mov eax, ebx
:00401912 89C3                 mov ebx, eax
:00401914 8B8700000000    mov eax, dword ptr [edi+00000000]
:0040191A 29C3                 sub ebx, eax
:0040191C 89D8                 mov eax, ebx
:0040191E 898708000000    mov dword ptr [edi+00000008], eax
:00401924 B800000000          mov eax, 00000000
:00401929 89872C000000    mov dword ptr [edi+0000002C], eax
:0040192F 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401935 89C3                 mov ebx, eax
:00401937 B800000000          mov eax, 00000000
:0040193C 39C3                 cmp ebx, eax
:0040193E 0F8D1C000000    jnl 00401960
:00401944 B801000000          mov eax, 00000001
:00401949 F7D8                 neg eax
:0040194B 89872C000000    mov dword ptr [edi+0000002C], eax
:00401951 B8351F4000          mov eax, 00401F35
:00401956 E885F8FFFF          call 004011E0
:0040195B E8B0F8FFFF          call 00401210

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040193E(C)

```

|
:00401960 E959FEFFFF          jmp 004017BE

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004017CD(C)

```
|
:00401965 8B8708000000    mov eax, dword ptr [edi+00000008]
:0040196B 89C3                mov ebx, eax
:0040196D B864000000    mov eax, 00000064
:00401972 93                xchg eax,ebx
:00401973 99                cdq
:00401974 F7FB                idiv ebx
:00401976 93                xchg eax,ebx
:00401977 89D8                mov eax, ebx
:00401979 898714000000    mov dword ptr [edi+00000014], eax
:0040197F 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401985 89C3                mov ebx, eax
:00401987 53                push ebx
:00401988 8B8714000000    mov eax, dword ptr [edi+00000014]
:0040198E 89C3                mov ebx, eax
:00401990 B864000000    mov eax, 00000064
:00401995 0FAFD8            imul ebx, eax
:00401998 89D8                mov eax, ebx
:0040199A 5B                pop ebx
:0040199B 29C3                sub ebx, eax
:0040199D 89D8                mov eax, ebx
:0040199F 898718000000    mov dword ptr [edi+00000018], eax
:004019A5 B8581F4000    mov eax, 00401F58
:004019AA E831F8FFFF        call 004011E0
:004019AF 8B8714000000    mov eax, dword ptr [edi+00000014]
:004019B5 E866F8FFFF        call 00401220
:004019BA B8651F4000    mov eax, 00401F65
:004019BF E81CF8FFFF        call 004011E0
:004019C4 8B8718000000    mov eax, dword ptr [edi+00000018]
:004019CA 89C3                mov ebx, eax
:004019CC B80A000000    mov eax, 0000000A
:004019D1 39C3                cmp ebx, eax
:004019D3 0F8D0A000000    jnl 004019E3
:004019D9 B8671F4000    mov eax, 00401F67
:004019DE E8FDF7FFFF        call 004011E0
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004019D3(C)

```
|
:004019E3 8B8718000000    mov eax, dword ptr [edi+00000018]
:004019E9 E832F8FFFF        call 00401220
:004019EE E81DF8FFFF        call 00401210
:004019F3 B8691F4000    mov eax, 00401F69
```

```

:004019F8 E8E3F7FFFF      call 004011E0
:004019FD E80EF8FFFF      call 00401210
:00401A02 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401A08 89C3              mov ebx, eax
:00401A0A B888130000      mov eax, 00001388
:00401A0F 39C3              cmp ebx, eax
:00401A11 0F8C26000000      jnl 00401A3D
:00401A17 B86A1F4000      mov eax, 00401F6A
:00401A1C E8BFF7FFFF      call 004011E0
:00401A21 E8EAF7FFFF      call 00401210
:00401A26 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401A2C 89C3              mov ebx, eax
:00401A2E B888130000      mov eax, 00001388
:00401A33 29C3              sub ebx, eax
:00401A35 89D8              mov eax, ebx
:00401A37 898708000000    mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401A11(C)

|

```

:00401A3D 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401A43 89C3              mov ebx, eax
:00401A45 B8D0070000      mov eax, 000007D0
:00401A4A 39C3              cmp ebx, eax
:00401A4C 0F8C94000000      jnl 00401AE6
:00401A52 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401A58 89C3              mov ebx, eax
:00401A5A B8D0070000      mov eax, 000007D0
:00401A5F 93                xchg eax,ebx
:00401A60 99                cdq
:00401A61 F7FB              idiv ebx
:00401A63 93                xchg eax,ebx
:00401A64 89D8              mov eax, ebx
:00401A66 89C3              mov ebx, eax
:00401A68 B801000000      mov eax, 00000001
:00401A6D 39C3              cmp ebx, eax
:00401A6F 0F850F000000      jne 00401A84
:00401A75 B87B1F4000      mov eax, 00401F7B
:00401A7A E861F7FFFF      call 004011E0
:00401A7F E88CF7FFFF      call 00401210

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401A6F(C)

|

```

:00401A84 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401A8A 89C3              mov ebx, eax

```

```

:00401A8C B8D0070000      mov eax, 000007D0
:00401A91 93                xchg eax,ebx
:00401A92 99                cdq
:00401A93 F7FB          idiv ebx
:00401A95 93                xchg eax,ebx
:00401A96 89D8          mov eax, ebx
:00401A98 89C3          mov ebx, eax
:00401A9A B801000000      mov eax, 00000001
:00401A9F 39C3          cmp ebx, eax
:00401AA1 0F840F000000      je 00401AB6
:00401AA7 B88C1F4000      mov eax, 00401F8C
:00401AAC E82FF7FFFF      call 004011E0
:00401AB1 E85AF7FFFF      call 00401210

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401AA1 (C)

```

|
:00401AB6 8B8708000000      mov eax, dword ptr [edi+00000008]
:00401ABC 89C3          mov ebx, eax
:00401ABE 53                push ebx
:00401ABF 8B8708000000      mov eax, dword ptr [edi+00000008]
:00401AC5 89C3          mov ebx, eax
:00401AC7 B8D0070000      mov eax, 000007D0
:00401ACC 93                xchg eax,ebx
:00401ACD 99                cdq
:00401ACE F7FB          idiv ebx
:00401AD0 93                xchg eax,ebx
:00401AD1 B8D0070000      mov eax, 000007D0
:00401AD6 0FAFD8          imul ebx, eax
:00401AD9 89D8          mov eax, ebx
:00401ADB 5B                pop ebx
:00401ADC 29C3          sub ebx, eax
:00401ADE 89D8          mov eax, ebx
:00401AE0 898708000000      mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401A4C (C)

```

|
:00401AE6 8B8708000000      mov eax, dword ptr [edi+00000008]
:00401AEC 89C3          mov ebx, eax
:00401AEE B8E8030000      mov eax, 000003E8
:00401AF3 39C3          cmp ebx, eax
:00401AF5 0F8C26000000      jl 00401B21
:00401AFB B89E1F4000      mov eax, 00401F9E
:00401B00 E8DBF6FFFF      call 004011E0
:00401B05 E806F7FFFF      call 00401210

```

```

:00401B0A 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401B10 89C3                mov ebx, eax
:00401B12 B8E8030000        mov eax, 000003E8
:00401B17 29C3                sub ebx, eax
:00401B19 89D8                mov eax, ebx
:00401B1B 898708000000        mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401AF5(C)

```

|
:00401B21 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401B27 89C3                mov ebx, eax
:00401B29 B8F4010000        mov eax, 000001F4
:00401B2E 39C3                cmp ebx, eax
:00401B30 0F8C26000000        jl 00401B5C
:00401B36 B8AF1F4000        mov eax, 00401FAF
:00401B3B E8A0F6FFFF        call 004011E0
:00401B40 E8CBF6FFFF        call 00401210
:00401B45 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401B4B 89C3                mov ebx, eax
:00401B4D B8F4010000        mov eax, 000001F4
:00401B52 29C3                sub ebx, eax
:00401B54 89D8                mov eax, ebx
:00401B56 898708000000        mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401B30(C)

```

|
:00401B5C 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401B62 89C3                mov ebx, eax
:00401B64 B8C8000000        mov eax, 000000C8
:00401B69 39C3                cmp ebx, eax
:00401B6B 0F8C94000000        jl 00401C05
:00401B71 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401B77 89C3                mov ebx, eax
:00401B79 B8C8000000        mov eax, 000000C8
:00401B7E 93                xchg eax,ebx
:00401B7F 99                cdq
:00401B80 F7FB                idiv ebx
:00401B82 93                xchg eax,ebx
:00401B83 89D8                mov eax, ebx
:00401B85 89C3                mov ebx, eax
:00401B87 B801000000        mov eax, 00000001
:00401B8C 39C3                cmp ebx, eax
:00401B8E 0F850F000000        jne 00401BA3
:00401B94 B8BF1F4000        mov eax, 00401FBF

```

```
:00401B99 E842F6FFFF      call 004011E0
:00401B9E E86DF6FFFF      call 00401210
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401B8E(C)

```
|
:00401BA3 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401BA9 89C3                mov ebx, eax
:00401BAB B8C8000000    mov eax, 000000C8
:00401BB0 93                xchg eax,ebx
:00401BB1 99                cdq
:00401BB2 F7FB                idiv ebx
:00401BB4 93                xchg eax,ebx
:00401BB5 89D8                mov eax, ebx
:00401BB7 89C3                mov ebx, eax
:00401BB9 B801000000    mov eax, 00000001
:00401BBE 39C3                cmp ebx, eax
:00401BC0 0F840F000000    je 00401BD5
:00401BC6 B8CF1F4000    mov eax, 00401FCF
:00401BCB E810F6FFFF      call 004011E0
:00401BD0 E83BF6FFFF      call 00401210
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401BC0(C)

```
|
:00401BD5 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401BDB 89C3                mov ebx, eax
:00401BDD 53                push ebx
:00401BDE 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401BE4 89C3                mov ebx, eax
:00401BE6 B8C8000000    mov eax, 000000C8
:00401BEB 93                xchg eax,ebx
:00401BEC 99                cdq
:00401BED F7FB                idiv ebx
:00401BEF 93                xchg eax,ebx
:00401BF0 B8C8000000    mov eax, 000000C8
:00401BF5 0FAFD8            imul ebx, eax
:00401BF8 89D8                mov eax, ebx
:00401BFA 5B                pop ebx
:00401BFB 29C3                sub ebx, eax
:00401BFD 89D8                mov eax, ebx
:00401BFF 898708000000    mov dword ptr [edi+00000008], eax
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401B6B(C)

```
|
```

```

:00401C05 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401C0B 89C3             mov ebx, eax
:00401C0D B864000000      mov eax, 00000064
:00401C12 39C3             cmp ebx, eax
:00401C14 0F8C26000000    jl 00401C40
:00401C1A B8E01F4000      mov eax, 00401FE0
:00401C1F E8BCF5FFFF      call 004011E0
:00401C24 E8E7F5FFFF      call 00401210
:00401C29 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401C2F 89C3             mov ebx, eax
:00401C31 B864000000      mov eax, 00000064
:00401C36 29C3             sub ebx, eax
:00401C38 89D8             mov eax, ebx
:00401C3A 898708000000    mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401C14(C)

```

|
:00401C40 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401C46 89C3             mov ebx, eax
:00401C48 B832000000      mov eax, 00000032
:00401C4D 39C3             cmp ebx, eax
:00401C4F 0F8C26000000    jl 00401C7B
:00401C55 B8EF1F4000      mov eax, 00401FEF
:00401C5A E881F5FFFF      call 004011E0
:00401C5F E8ACF5FFFF      call 00401210
:00401C64 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401C6A 89C3             mov ebx, eax
:00401C6C B832000000      mov eax, 00000032
:00401C71 29C3             sub ebx, eax
:00401C73 89D8             mov eax, ebx
:00401C75 898708000000    mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401C4F(C)

```

|
:00401C7B 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401C81 89C3             mov ebx, eax
:00401C83 B819000000      mov eax, 00000019
:00401C88 39C3             cmp ebx, eax
:00401C8A 0F8C26000000    jl 00401CB6
:00401C90 B807204000      mov eax, 00402007
:00401C95 E846F5FFFF      call 004011E0
:00401C9A E871F5FFFF      call 00401210
:00401C9F 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401CA5 89C3             mov ebx, eax

```

```

:00401CA7 B819000000      mov eax, 00000019
:00401CAC 29C3             sub ebx, eax
:00401CAE 89D8             mov eax, ebx
:00401CB0 898708000000      mov dword ptr [edi+00000008], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401C8A(C)

|

```

:00401CB6 8B8708000000      mov eax, dword ptr [edi+00000008]
:00401CBC 89C3             mov ebx, eax
:00401CBE B80A000000      mov eax, 0000000A
:00401CC3 39C3             cmp ebx, eax
:00401CC5 0F8C94000000     jl 00401D5F
:00401CCB 8B8708000000      mov eax, dword ptr [edi+00000008]
:00401CD1 89C3             mov ebx, eax
:00401CD3 B80A000000      mov eax, 0000000A
:00401CD8 93              xchg eax,ebx
:00401CD9 99              cdq
:00401CDA F7FB             idiv ebx
:00401CDC 93              xchg eax,ebx
:00401CDD 89D8             mov eax, ebx
:00401CDF 89C3             mov ebx, eax
:00401CE1 B801000000      mov eax, 00000001
:00401CE6 39C3             cmp ebx, eax
:00401CE8 0F850F000000     jne 00401CFD
:00401CEE B81F204000      mov eax, 0040201F
:00401CF3 E8E8F4FFFF      call 004011E0
:00401CF8 E813F5FFFF      call 00401210

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401CE8(C)

|

```

:00401CFD 8B8708000000      mov eax, dword ptr [edi+00000008]
:00401D03 89C3             mov ebx, eax
:00401D05 B80A000000      mov eax, 0000000A
:00401D0A 93              xchg eax,ebx
:00401D0B 99              cdq
:00401D0C F7FB             idiv ebx
:00401D0E 93              xchg eax,ebx
:00401D0F 89D8             mov eax, ebx
:00401D11 89C3             mov ebx, eax
:00401D13 B801000000      mov eax, 00000001
:00401D18 39C3             cmp ebx, eax
:00401D1A 0F840F000000     je 00401D2F
:00401D20 B837204000      mov eax, 00402037
:00401D25 E8B6F4FFFF      call 004011E0

```



```
:00401D2A E8E1F4FFFF          call 00401210
```

```
* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401D1A(C)
```

```
|
```

```
:00401D2F 8B8708000000          mov eax, dword ptr [edi+00000008]
:00401D35 89C3                  mov ebx, eax
:00401D37 53                    push ebx
:00401D38 8B8708000000          mov eax, dword ptr [edi+00000008]
:00401D3E 89C3                  mov ebx, eax
:00401D40 B80A000000          mov eax, 0000000A
:00401D45 93                    xchg eax,ebx
:00401D46 99                    cdq
:00401D47 F7FB                  idiv ebx
:00401D49 93                    xchg eax,ebx
:00401D4A B80A000000          mov eax, 0000000A
:00401D4F 0FAFD8              imul ebx, eax
:00401D52 89D8                  mov eax, ebx
:00401D54 5B                    pop ebx
:00401D55 29C3                  sub ebx, eax
:00401D57 89D8                  mov eax, ebx
:00401D59 898708000000          mov dword ptr [edi+00000008], eax
```

```
* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401CC5(C)
```

```
|
```

```
:00401D5F 8B8708000000          mov eax, dword ptr [edi+00000008]
:00401D65 89C3                  mov ebx, eax
:00401D67 B805000000          mov eax, 00000005
:00401D6C 39C3                  cmp ebx, eax
:00401D6E 0F8C26000000          jl 00401D9A
:00401D74 B850204000          mov eax, 00402050
:00401D79 E862F4FFFF          call 004011E0
:00401D7E E88DF4FFFF          call 00401210
:00401D83 8B8708000000          mov eax, dword ptr [edi+00000008]
:00401D89 89C3                  mov ebx, eax
:00401D8B B805000000          mov eax, 00000005
:00401D90 29C3                  sub ebx, eax
:00401D92 89D8                  mov eax, ebx
:00401D94 898708000000          mov dword ptr [edi+00000008], eax
```

```
* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401D6E(C)
```

```
|
```

```
:00401D9A 8B8708000000          mov eax, dword ptr [edi+00000008]
:00401DA0 89C3                  mov ebx, eax
```

```
:00401DA2 B801000000      mov eax, 00000001
:00401DA7 39C3              cmp ebx, eax
:00401DA9 0F8E1A000000     jle 00401DC9
:00401DAF 8B8708000000     mov eax, dword ptr [edi+00000008]
:00401DB5 E866F4FFFF       call 00401220
:00401DBA B867204000       mov eax, 00402067
:00401DBF E81CF4FFFF       call 004011E0
:00401DC4 E847F4FFFF       call 00401210
```

```
* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401DA9(C)
```

```
|
:00401DC9 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401DCF 89C3            mov ebx, eax
:00401DD1 B801000000    mov eax, 00000001
:00401DD6 39C3            cmp ebx, eax
:00401DD8 0F850F000000    jne 00401DED
:00401DDE B87D204000    mov eax, 0040207D
:00401DE3 E8F8F3FFFF    call 004011E0
:00401DE8 E823F4FFFF    call 00401210
```

```
* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401DD8 (C)
```

```
|
:00401DED E996F5FFFF jmp 00401388
:00401DF2 00000000000000000000 BYTE 10 DUP(0)
:00401DFC 00000000000000000000 BYTE 10 DUP(0)
:00401E06 00000000000000000000 BYTE 10 DUP(0)
:00401E10 00000000000000000000 BYTE 10 DUP(0)
:00401E1A 00000000000000000000 BYTE 8 DUP(0)
```

```
:00401E22 56          push esi
:00401E23 41          inc ecx
:00401E24 4C          dec esp
:00401E25 4F          dec edi
:00401E26 52          push edx
:00401E27 204655     and byte ptr [esi+55], al
:00401E2A 45          inc ebp
:00401E2B 52          push edx
:00401E2C 41          inc ecx
:00401E2D 20444520   and byte ptr [ebp+2*eax+20], al
:00401E31 52          push edx
:00401E32 41          inc ecx
:00401E33 4E          dec esi
:00401E34 47          inc edi
```

```

:00401E35 4F      dec edi
:00401E36 2100     and dword ptr [eax], eax
:00401E38 2A2A     sub ch, byte ptr [edx]
:00401E3A 2A2A     sub ch, byte ptr [edx]
:00401E3C 2A2A     sub ch, byte ptr [edx]
:00401E3E 2A2A     sub ch, byte ptr [edx]
:00401E40 2A2A     sub ch, byte ptr [edx]
:00401E42 2A2A     sub ch, byte ptr [edx]
:00401E44 2A2A     sub ch, byte ptr [edx]
:00401E46 2A2A     sub ch, byte ptr [edx]
:00401E48 2A2A     sub ch, byte ptr [edx]
:00401E4A 2A2A     sub ch, byte ptr [edx]
:00401E4C 2A2A     sub ch, byte ptr [edx]
:00401E4E 2A2A     sub ch, byte ptr [edx]
:00401E50 2A2A     sub ch, byte ptr [edx]
:00401E52 2A2A     sub ch, byte ptr [edx]
:00401E54 2A2A     sub ch, byte ptr [edx]
:00401E56 2A2A     sub ch, byte ptr [edx]
:00401E58 2A2A     sub ch, byte ptr [edx]
:00401E5A 2A2A     sub ch, byte ptr [edx]
:00401E5C 2A2A     sub ch, byte ptr [edx]
:00401E5E 2A2A     sub ch, byte ptr [edx]
:00401E60 2A2A     sub ch, byte ptr [edx]
:00401E62 2A00     sub al, byte ptr [eax]
:00401E64 56      push esi
:00401E65 55      push ebp
:00401E66 45      inc ebp
:00401E67 4C      dec esp
:00401E68 54      push esp
:00401E69 4F      dec edi
:00401E6A 205041  and byte ptr [eax+41], dl
:00401E6D 52      push edx
:00401E6E 41      inc ecx
:00401E6F 20494D  and byte ptr [ecx+4D], cl
:00401E72 50      push eax
:00401E73 4F      dec edi
:00401E74 52      push edx
:00401E75 54      push esp
:00401E76 45      inc ebp
:00401E77 53      push ebx
:00401E78 205041  and byte ptr [eax+41], dl
:00401E7B 47      inc edi
:00401E7C 41      inc ecx
:00401E7D 44      inc esp
:00401E7E 4F      dec edi
:00401E7F 53      push ebx

```

```

:00401E80 20434F      and byte ptr [ebx+4F], al
:00401E83 4E             dec esi
:00401E84 20554E      and byte ptr [ebp+4E], dl
:00401E87 204249      and byte ptr [edx+49], al
:00401E8A 4C           dec esp
:00401E8B 4C           dec esp
:00401E8C 45           inc ebp
:00401E8D 54           push esp
:00401E8E 45           inc ebp
:00401E8F 002A        add byte ptr [edx], ch
:00401E91 2A2A        sub ch, byte ptr [edx]
:00401E93 2A2A        sub ch, byte ptr [edx]
:00401E95 2A2A        sub ch, byte ptr [edx]
:00401E97 2A2A        sub ch, byte ptr [edx]
:00401E99 2A2A        sub ch, byte ptr [edx]
:00401E9B 2A2A        sub ch, byte ptr [edx]
:00401E9D 2A2A        sub ch, byte ptr [edx]
:00401E9F 2A2A        sub ch, byte ptr [edx]
:00401EA1 2A2A        sub ch, byte ptr [edx]
:00401EA3 2A2A        sub ch, byte ptr [edx]
:00401EA5 2A2A        sub ch, byte ptr [edx]
:00401EA7 2A2A        sub ch, byte ptr [edx]
:00401EA9 2A2A        sub ch, byte ptr [edx]
:00401EAB 2A2A        sub ch, byte ptr [edx]
:00401EAD 2A2A        sub ch, byte ptr [edx]
:00401EAF 2A2A        sub ch, byte ptr [edx]
:00401EB1 2A2A        sub ch, byte ptr [edx]
:00401EB3 2A2A        sub ch, byte ptr [edx]
:00401EB5 2A2A        sub ch, byte ptr [edx]
:00401EB7 2A2A        sub ch, byte ptr [edx]
:00401EB9 2A2A        sub ch, byte ptr [edx]
:00401EBB 0000      add byte ptr [eax], al
:00401EBD 49           dec ecx
:00401EBE 4D           dec ebp
:00401EBF 50           push eax
:00401EC0 4F           dec edi
:00401EC1 52           push edx
:00401EC2 54           push esp
:00401EC3 45           inc ebp
:00401EC4 2028        and byte ptr [eax], ch
:00401EC6 6D           insd
:00401EC7 696E2024302E30
302E3024      imul ebp, dword ptr [esi+20],
:00401ECE 3120        xor dword ptr [eax], esp
:00401ED0 7920        jns 00401EF2
:00401ED2 6D           insd

```

```

:00401ED3 61          popad
:00401ED4 7820         js 00401EF6
:00401ED6 2431         and al, 31
:00401ED8 3030         xor byte ptr [eax], dh
:00401EDA 2E3030       xor byte ptr cs:[eax], dh
:00401EDD 2900         sub dword ptr [eax], eax
:00401EDF 43          inc ebx
:00401EE0 45          inc ebp
:00401EE1 4E          dec esi
:00401EE2 54          push esp
:00401EE3 41          inc ecx
:00401EE4 56          push esi
:00401EE5 4F          dec edi
:00401EE6 53          push ebx
:00401EE7 3A20         cmp ah, byte ptr [eax]
:00401EE9 005045       add byte ptr [eax+45], dl
:00401EEC 53          push ebx
:00401EED 4F          dec edi
:00401EEE 53          push ebx
:00401EEF 3A20         cmp ah, byte ptr [eax]
:00401EF1 00494D       add byte ptr [ecx+4D], cl
:00401EF4 50          push eax
:00401EF5 4F          dec edi

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

| :00401ED4 (C)

```

|
:00401EF6 52          push edx
:00401EF7 54          push esp
:00401EF8 45          inc ebp
:00401EF9 3A20         cmp ah, byte ptr [eax]
:00401EFB 2400         and al, 00
:00401EFD 2E0030       add byte ptr cs:[eax], dh
:00401F00 004249       add byte ptr [edx+49], al
:00401F03 4C          dec esp
:00401F04 4C          dec esp
:00401F05 45          inc ebp
:00401F06 54          push esp
:00401F07 45          inc ebp
:00401F08 2028         and byte ptr [eax], ch
:00401F0A 6D          insd
:00401F0B 696E2024322079
79203224      imul ebp, dword ptr [esi+20],
:00401F12 206D61       and byte ptr [ebp+61], ch
:00401F15 7820         js 00401F37
:00401F17 2431         and al, 31

```

```

:00401F19 3030          xor byte ptr [eax], dh
:00401F1B 293A          sub dword ptr [edx], edi
:00401F1D 202400       and byte ptr [eax+eax], ah
:00401F20 42           inc edx
:00401F21 49           dec ecx
:00401F22 4C           dec esp
:00401F23 4C           dec esp
:00401F24 45           inc ebp
:00401F25 54           push esp
:00401F26 45           inc ebp
:00401F27 20494E       and byte ptr [ecx+4E], cl
:00401F2A 45           inc ebp
:00401F2B 58           pop eax
:00401F2C 49           dec ecx
:00401F2D 53           push ebx
:00401F2E 54           push esp
:00401F2F 45           inc ebp
:00401F30 4E           dec esi
:00401F31 54           push esp
:00401F32 45           inc ebp
:00401F33 2100        and dword ptr [eax], eax
:00401F35 42           inc edx
:00401F36 49           dec ecx

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

| :00401F15(C)

|

```

:00401F37 4C           dec esp
:00401F38 4C           dec esp
:00401F39 45           inc ebp
:00401F3A 54           push esp
:00401F3B 45           inc ebp
:00401F3C 20494E       and byte ptr [ecx+4E], cl
:00401F3F 53           push ebx
:00401F40 55           push ebp
:00401F41 46           inc esi
:00401F42 49           dec ecx
:00401F43 43           inc ebx
:00401F44 49           dec ecx
:00401F45 45           inc ebp
:00401F46 4E           dec esi
:00401F47 54           push esp
:00401F48 45           inc ebp
:00401F49 205041       and byte ptr [eax+41], dl
:00401F4C 52           push edx
:00401F4D 41           inc ecx

```

```

:00401F4E 20454C      and byte ptr [ebp+4C], al
:00401F51 205041      and byte ptr [eax+41], dl
:00401F54 47          inc edi
:00401F55 4F          dec edi
:00401F56 2100      and dword ptr [eax], eax
:00401F58 53          push ebx
:00401F59 7520      jne 00401F7B
:00401F5B 7675      jbe 00401FD2
:00401F5D 65          BYTE 065h

```

```

:00401F5E 6C          insb
:00401F5F 746F      je 00401FD0
:00401F61 3A20      cmp ah, byte ptr [eax]
:00401F63 2400      and al, 00
:00401F65 2E0030    add byte ptr cs:[eax], dh
:00401F68 0000      add byte ptr [eax], al
:00401F6A 3120      xor dword ptr [eax], esp
:00401F6C 62696C    bound ebp, dword ptr [ecx+6C]
:00401F6F 6C          insb
:00401F70 65          BYTE 065h

```

```

:00401F71 7465      je 00401FD8
:00401F73 20646520 and byte ptr [ebp+20], ah
:00401F77 2435      and al, 35
:00401F79 3000      xor byte ptr [eax], al

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401F59(C)

```

|
:00401F7B 3120      xor dword ptr [eax], esp
:00401F7D 62696C    bound ebp, dword ptr [ecx+6C]
:00401F80 6C          insb
:00401F81 65          BYTE 065h

```

```

:00401F82 7465      je 00401FE9
:00401F84 20646520 and byte ptr [ebp+20], ah
:00401F88 2432      and al, 32
:00401F8A 3000      xor byte ptr [eax], al
:00401F8C 3220      xor ah, byte ptr [eax]
:00401F8E 62696C    bound ebp, dword ptr [ecx+6C]
:00401F91 6C          insb
:00401F92 65          BYTE 065h

```

```

:00401F93 7465          je 00401FFA
:00401F95 7320          jnb 00401FB7
:00401F97 64           BYTE 064h

:00401F98 65202432      and byte ptr gs:[edx+esi], ah
:00401F9C 3000          xor byte ptr [eax], al
:00401F9E 3120          xor dword ptr [eax], esp
:00401FA0 62696C        bound ebp, dword ptr [ecx+6C]
:00401FA3 6C           insb
:00401FA4 65           BYTE 065h

:00401FA5 7465          je 0040200C
:00401FA7 20646520      and byte ptr [ebp+20], ah
:00401FAB 2431          and al, 31
:00401FAD 3000          xor byte ptr [eax], al
:00401FAF 3120          xor dword ptr [eax], esp
:00401FB1 62696C        bound ebp, dword ptr [ecx+6C]
:00401FB4 6C           insb
:00401FB5 65           BYTE 065h

:00401FB6 7465          je 0040201D
:00401FB8 20646520      and byte ptr [ebp+20], ah
:00401FBC 2435          and al, 35
:00401FBE 0031          add byte ptr [ecx], dh
:00401FC0 206269        and byte ptr [edx+69], ah
:00401FC3 6C           insb
:00401FC4 6C           insb
:00401FC5 65           BYTE 065h

:00401FC6 7465          je 0040202D
:00401FC8 20646520      and byte ptr [ebp+20], ah
:00401FCC 2432          and al, 32
:00401FCE 0032          add byte ptr [edx], dh

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401F5F(C)

|

```

:00401FD0 206269        and byte ptr [edx+69], ah
:00401FD3 6C           insb
:00401FD4 6C           insb
:00401FD5 65           BYTE 065h

```



```
:00401FD6 7465                                je 0040203D
```

```
* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401F71(C)
```

```
|
```

```
:00401FD8 7320                                jnb 00401FFA
```

```
:00401FDA 64                                BYTE 064h
```

```
:00401FDB 65202432                          and byte ptr gs:[edx+esi], ah
```

```
:00401FDF 0031                          add byte ptr [ecx], dh
```

```
:00401FE1 206D6F                          and byte ptr [ebp+6F], ch
```

```
:00401FE4 6E                          outsb
```

```
:00401FE5 65                          BYTE 065h
```

```
:00401FE6 64                          BYTE 064h
```

```
:00401FE7 61                          popad
```

```
:00401FE8 20646520                      and byte ptr [ebp+20], ah
```

```
:00401FEC 2431                          and al, 31
```

```
:00401FEE 0031                          add byte ptr [ecx], dh
```

```
:00401FF0 206D6F                          and byte ptr [ebp+6F], ch
```

```
:00401FF3 6E                          outsb
```

```
:00401FF4 65                          BYTE 065h
```

```
:00401FF5 64                          BYTE 064h
```

```
:00401FF6 61                          popad
```

```
:00401FF7 20646520                      and byte ptr [ebp+20], ah
```

```
:00401FFB 3530206365                    xor eax, 65632030
```

```
:00402000 6E                          outsb
```

```
:00402001 7461                          je 00402064
```

```
:00402003 766F                          jbe 00402074
```

```
:00402005 7300                          jnb 00402007
```

```
* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00402005(C)
```

```
|
```

```
:00402007 3120                          xor dword ptr [eax], esp
```

```
:00402009 6D                          insd
```

```
:0040200A 6F          outsd
:0040200B 6E          outsb
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401FA5(C)

```
|
:0040200C 65          BYTE 065h
:0040200D 64          BYTE 064h
```

```
:0040200E 61          popad
:0040200F 20646520    and byte ptr [ebp+20], ah
:00402013 32352063656E xor dh, byte ptr [6E656320]
:00402019 7461        je 0040207C
:0040201B 766F        jbe 0040208C
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401FB6(C)

```
|
:0040201D 7300        jnb 0040201F
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040201D(C)

```
|
:0040201F 3120        xor dword ptr [eax], esp
:00402021 6D          insd
:00402022 6F          outsd
:00402023 6E          outsb
:00402024 65          BYTE 065h
```

```
:00402025 64          BYTE 064h
```

```
:00402026 61          popad
:00402027 20646520    and byte ptr [ebp+20], ah
:0040202B 3130        xor dword ptr [eax], esi
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401FC6(C)

```
|
:0040202D 206365      and byte ptr [ebx+65], ah
:00402030 6E          outsb
:00402031 7461        je 00402094
:00402033 766F        jbe 004020A4
:00402035 7300        jnb 00402037
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00402035(C)

```
|
:00402037 3220          xor ah, byte ptr [eax]
:00402039 6D             insd
:0040203A 6F             outsd
:0040203B 6E             outsb
:0040203C 65             BYTE 065h
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401FD6(C)

```
|
:0040203D 64             BYTE 064h
:0040203E 61             popad
:0040203F 7320          jnb 00402061
:00402041 64             BYTE 064h
```

```
:00402042 652031        and byte ptr gs:[ecx], dh
:00402045 3020          xor byte ptr [eax], ah
:00402047 63656E        arpl dword ptr [ebp+6E], esp
:0040204A 7461          je 004020AD
:0040204C 766F          jbe 004020BD
:0040204E 7300          jnb 00402050
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040204E(C)

```
|
:00402050 3120          xor dword ptr [eax], esp
:00402052 6D             insd
:00402053 6F             outsd
:00402054 6E             outsb
:00402055 65             BYTE 065h
```

```
:00402056 64             BYTE 064h
```

```
:00402057 61             popad
:00402058 20646520      and byte ptr [ebp+20], ah
:0040205C 352063656E    xor eax, 6E656320
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:0040203F(C)

|

```
:00402061 7461          je 004020C4
:00402063 766F          jbe 004020D4
:00402065 7300          jnb 00402067
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00402065(C)

|

```
:00402067 206D6F        and byte ptr [ebp+6F], ch
:0040206A 6E            outsb
:0040206B 65            BYTE 065h
```

```
:0040206C 64            BYTE 064h
```

```
:0040206D 61            popad
:0040206E 7320          jnb 00402090
:00402070 64            BYTE 064h
```

```
:00402071 652031        and byte ptr gs:[ecx], dh
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00402003(C)

|

```
:00402074 206365        and byte ptr [ebx+65], ah
:00402077 6E            outsb
:00402078 7461          je 004020DB
:0040207A 766F          jbe 004020EB
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00402019(C)

|

```
:0040207C 0031          add byte ptr [ecx], dh
:0040207E 206D6F        and byte ptr [ebp+6F], ch
:00402081 6E            outsb
:00402082 65            BYTE 065h
```

```
:00402083 64            BYTE 064h
```

```
:00402084 61            popad
:00402085 20646520      and byte ptr [ebp+20], ah
```

```
:00402089 3120          xor dword ptr [eax], esp
:0040208B 63656E        arpl dword ptr [ebp+6E], esp
:0040208E 7461          je 004020F1
```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:0040206E(C)

|

```
:00402090 766F          jbe 00402101
:00402092 00000000000000000000000000000000 BYTE 10 DUP(0)
```

07

```

CONST N = 20;

VAR A, B, C;

PROCEDURE TRIANGULO;
VAR A, B;
BEGIN
  WRITELN;
  A := 15;
  WHILE A > 0 DO
    BEGIN
      B := 0;
      WHILE B < A DO
        BEGIN
          WRITE ('*');
          B := B + 1
        END;
      WRITELN;
      A := A - 1;
    END
  END;
END;

BEGIN
  A := 1;
  WHILE A <= N DO
    BEGIN
      WRITE (A, ' ');
      A := A + 1
    END;

    CALL TRIANGULO;

    B := -N;
    C := 0;
    WHILE B < C DO
      BEGIN
        WRITE (B, ' ');
        B := B + 1
      END;
      WRITELN;
    END.

```



```

000003d0    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
000003e0    60 89 c6 30 c0 02 06 74    08 46 e8 e1 fe ff ff eb    `..0...t.F.....
000003f0    f2 61 90 c3 00 00 00 00    00 00 00 00 00 00 00 00    .a.....
00000400    04 30 e8 c9 fe ff ff c3    00 00 00 00 00 00 00 00    .0.....
00000410    b0 0d e8 b9 fe ff ff b0    0a e8 b2 fe ff ff c3 00    .....
00000420    3d 00 00 00 80 75 4e b0    2d e8 a2 fe ff ff b0 02    =....uN.-.....
00000430    e8 cb ff ff ff b0 01 e8    c4 ff ff ff b0 04 e8 bd    .....
00000440    ff ff ff b0 07 e8 b6 ff    ff ff b0 04 e8 af ff ff    .....
00000450    ff b0 08 e8 a8 ff ff ff    b0 03 e8 a1 ff ff ff b0    .....
00000460    06 e8 9a ff ff ff b0 04    e8 93 ff ff ff b0 08 e8    .....
00000470    8c ff ff ff c3 3d 00 00    00 00 7d 0b 50 b0 2d e8    .....=....}.P.-.
00000480    4c fe ff ff 58 f7 d8 3d    0a 00 00 00 0f 8c ef 00    L...X..=.....
00000490    00 00 3d 64 00 00 00 0f    8c d1 00 00 00 3d e8 03    ..=d.....=.
000004a0    00 00 0f 8c b3 00 00 00    3d 10 27 00 00 0f 8c 95    .....=.'.....
000004b0    00 00 00 3d a0 86 01 00    7c 7b 3d 40 42 0f 00 7c    ...=....|{@B..|
000004c0    61 3d 80 96 98 00 7c 47    3d 00 e1 f5 05 7c 2d 3d    a=....|G=....|-=
000004d0    00 ca 9a 3b 7c 13 ba 00    00 00 00 bb 00 ca 9a 3b    ...;|.....;
000004e0    f7 fb 52 e8 18 ff ff ff    58 ba 00 00 00 00 bb 00    ..R....X.....
000004f0    e1 f5 05 f7 fb 52 e8 05    ff ff ff 58 ba 00 00 00    .....R....X....
00000500    00 bb 80 96 98 00 f7 fb    52 e8 f2 fe ff ff 58 ba    .....R....X.
00000510    00 00 00 00 bb 40 42 0f    00 f7 fb 52 e8 df fe ff    .....@B....R....
00000520    ff 58 ba 00 00 00 00 bb    a0 86 01 00 f7 fb 52 e8    .X.....R.
00000530    cc fe ff ff 58 ba 00 00    00 00 bb 10 27 00 00 f7    ....X.....'...
00000540    fb 52 e8 b9 fe ff ff 58    ba 00 00 00 00 bb e8 03    .R....X.....
00000550    00 00 f7 fb 52 e8 a6 fe    ff ff 58 ba 00 00 00 00    ....R....X.....
00000560    bb 64 00 00 00 f7 fb 52    e8 93 fe ff ff 58 ba 00    .d....R....X..
00000570    00 00 00 bb 0a 00 00 00    f7 fb 52 e8 80 fe ff ff    .....R.....
00000580    58 e8 7a fe ff ff c3 00    ff 15 00 10 40 00 00 00    X.z.....@...
00000590    b9 00 00 00 00 b3 03 51    53 e8 a2 fd ff ff 5b 59    .....QS.....[Y
000005a0    3c 0d 0f 84 34 01 00 00    3c 08 0f 84 94 00 00 00    <...4...<.....
000005b0    3c 2d 0f 84 09 01 00 00    3c 30 7c db 3c 39 7f d7    <-.....<0|<9..
000005c0    2c 30 80 fb 00 74 d0 80    fb 02 75 0c 81 f9 00 00    ,0...t....u....
000005d0    00 00 75 04 3c 00 74 bf    80 fb 03 75 0a 3c 00 75    ..u.<.t....u.<.u
000005e0    04 b3 00 eb 02 b3 01 81    f9 cc cc cc 0c 7f a8 81    .....
000005f0    f9 34 33 33 f3 7c a0 88    c7 b8 0a 00 00 00 f7 e9    .433.|.....
00000600    3d 08 00 00 80 74 11 3d    f8 ff ff 7f 75 13 80 ff    =....t.=....u...
00000610    07 7e 0e e9 7f ff ff ff    80 ff 08 0f 8f 76 ff ff    .~.....v...
00000620    ff b9 00 00 00 00 88 f9    80 fb 02 74 04 01 c1 eb    .....t....
00000630    03 29 c8 91 88 f8 51 53    e8 c3 fd ff ff 5b 59 e9    .)....QS.....[Y.
00000640    53 ff ff ff 80 fb 03 0f    84 4a ff ff ff 51 53 b0    S.....J...QS.
00000650    08 e8 7a fc ff ff b0 20    e8 73 fc ff ff b0 08 e8    ..z....s.....
00000660    6c fc ff ff 5b 59 80 fb    00 75 07 b3 03 e9 25 ff    l...[Y...u...%.
00000670    ff ff 80 fb 02 75 0f 81    f9 00 00 00 00 75 07 b3    .....u.....u..
00000680    03 e9 11 ff ff ff 89 c8    b9 0a 00 00 00 ba 00 00    .....
00000690    00 00 3d 00 00 00 00 7d    08 f7 d8 f7 f9 f7 d8 eb    ..=....}).....
000006a0    02 f7 f9 89 c1 81 f9 00    00 00 00 0f 85 e6 fe ff    .....
000006b0    ff 80 fb 02 0f 84 dd fe    ff ff b3 03 e9 d6 fe ff    .....
000006c0    ff 80 fb 03 0f 85 cd fe    ff ff b0 2d 51 53 e8 fd    .....-QS..
000006d0    fb ff ff 5b 59 b3 02 e9    bb fe ff ff 80 fb 03 0f    ...[Y.....
000006e0    84 b2 fe ff ff 80 fb 02    75 0c 81 f9 00 00 00 00    .....u.....
000006f0    0f 84 a1 fe ff ff 51 e8    14 fd ff ff 59 89 c8 c3    .....Q.....Y...
00000700    bf 6c 16 40 00 e9 9f 00    00 00 b8 80 16 40 00 e8    .l.@.....@..
00000710    cc fc ff ff e8 f7 fc ff    ff b8 0f 00 00 00 89 87    .....
00000720    0c 00 00 00 8b 87 0c 00    00 00 89 c3 b8 00 00 00    .....
00000730    00 39 c3 0f 8e 6f 00 00    00 b8 00 00 00 00 89 87    .9...o.....
00000740    10 00 00 00 8b 87 10 00    00 00 89 c3 8b 87 0c 00    .....
00000750    00 00 39 c3 0f 8d 23 00    00 00 b8 81 16 40 00 e8    ..9...#.....@..
00000760    7c fc ff ff 8b 87 10 00    00 00 89 c3 b8 01 00 00    |.....
00000770    00 01 c3 89 d8 89 87 10    00 00 00 eb c7 b8 83 16    .....
00000780    40 00 e8 59 fc ff ff e8    84 fc ff ff 8b 87 0c 00    @..Y.....
00000790    00 00 89 c3 b8 01 00 00    00 29 c3 89 d8 89 87 0c    .....).....
000007a0    00 00 00 e9 7c ff ff ff    c3 b8 01 00 00 00 89 87    ....|.....

```


000007b0	00 00 00 00 8b 87 00 00	00 00 89 c3 b8 14 00 00
000007c0	00 39 c3 0f 8f 2e 00 00	00 8b 87 00 00 00 00 e8	.9.....
000007d0	4c fc ff ff b8 84 16 40	00 e8 02 fc ff ff 8b 87	L.....@.....
000007e0	00 00 00 00 89 c3 b8 01	00 00 00 01 c3 89 d8 89
000007f0	87 00 00 00 00 eb bd e8	0e ff ff ff b8 14 00 00
00000800	00 f7 d8 89 87 04 00 00	00 b8 00 00 00 00 89 87
00000810	08 00 00 00 8b 87 04 00	00 00 89 c3 8b 87 08 00
00000820	00 00 39 c3 0f 8d 2e 00	00 00 8b 87 04 00 00 00	..9.....
00000830	e8 eb fb ff ff b8 86 16	40 00 e8 a1 fb ff ff 8b@.....
00000840	87 04 00 00 00 89 c3 b8	01 00 00 00 01 c3 89 d8
00000850	89 87 04 00 00 00 eb bc	b8 88 16 40 00 e8 7e fb@...~.
00000860	ff ff e8 a9 fb ff ff e9	1c fd ff ff 00 00 00 00
00000870	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
00000880	00 2a 00 00 20 00 20 00	00 00 00 00 00 00 00 00	.*.
00000890	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008a0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008b0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008c0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008d0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008e0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000008f0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

desensamblado

```

:00401500 BF6C164000      mov edi, 0040166C
:00401505 E99F000000      jmp 004015A9
:0040150A B880164000      mov eax, 00401680
:0040150F E8CCFCFFFF      call 004011E0
:00401514 E8F7FCFFFF      call 00401210
:00401519 B80F000000      mov eax, 0000000F
:0040151E 89870C000000      mov dword ptr [edi+0000000C], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004015A3(U)

```

|
:00401524 8B870C000000      mov eax, dword ptr [edi+0000000C]
:0040152A 89C3              mov ebx, eax
:0040152C B800000000      mov eax, 00000000
:00401531 39C3              cmp ebx, eax
:00401533 0F8E6F000000      jle 004015A8
:00401539 B800000000      mov eax, 00000000
:0040153E 898710000000      mov dword ptr [edi+00000010], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040157B(U)

```

|
:00401544 8B8710000000      mov eax, dword ptr [edi+00000010]
:0040154A 89C3              mov ebx, eax
:0040154C 8B870C000000      mov eax, dword ptr [edi+0000000C]
:00401552 39C3              cmp ebx, eax
:00401554 0F8D23000000      jnl 0040157D
:0040155A B881164000      mov eax, 00401681
:0040155F E87CFCFFFF      call 004011E0
:00401564 8B8710000000      mov eax, dword ptr [edi+00000010]
:0040156A 89C3              mov ebx, eax
:0040156C B801000000      mov eax, 00000001
:00401571 01C3              add ebx, eax
:00401573 89D8              mov eax, ebx
:00401575 898710000000      mov dword ptr [edi+00000010], eax
:0040157B EBC7              jmp 00401544

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401554(C)

```

|
:0040157D B883164000      mov eax, 00401683
:00401582 E859FCFFFF      call 004011E0
:00401587 E884FCFFFF      call 00401210
:0040158C 8B870C000000      mov eax, dword ptr [edi+0000000C]

```

```

:00401592 89C3                mov ebx, eax
:00401594 B801000000           mov eax, 00000001
:00401599 29C3                sub ebx, eax
:0040159B 89D8                mov eax, ebx
:0040159D 89870C000000        mov dword ptr [edi+0000000C], eax
:004015A3 E97CFFFFFF          jmp 00401524

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401533(C)

```

|
:004015A8 C3                ret

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401505(U)

```

|
:004015A9 B801000000           mov eax, 00000001
:004015AE 898700000000        mov dword ptr [edi+00000000], eax

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004015F5(U)

```

|
:004015B4 8B8700000000        mov eax, dword ptr [edi+00000000]
:004015BA 89C3                mov ebx, eax
:004015BC B814000000        mov eax, 00000014
:004015C1 39C3                cmp ebx, eax
:004015C3 0F8F2E000000        jg 004015F7
:004015C9 8B8700000000        mov eax, dword ptr [edi+00000000]
:004015CF E84CFCFFFFFF          call 00401220
:004015D4 B884164000        mov eax, 00401684
:004015D9 E802FCFFFFFF          call 004011E0
:004015DE 8B8700000000        mov eax, dword ptr [edi+00000000]
:004015E4 89C3                mov ebx, eax
:004015E6 B801000000        mov eax, 00000001
:004015EB 01C3                add ebx, eax
:004015ED 89D8                mov eax, ebx
:004015EF 898700000000        mov dword ptr [edi+00000000], eax
:004015F5 EBBD                jmp 004015B4

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:004015C3(C)

```

|
:004015F7 E80EFFFFFF          call 0040150A
:004015FC B814000000        mov eax, 00000014
:00401601 F7D8                neg eax
:00401603 898704000000        mov dword ptr [edi+00000004], eax
:00401609 B800000000        mov eax, 00000000

```

```
:0040160E 898708000000    mov dword ptr [edi+00000008], eax
```

```
* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401656(U)
```

1

```
:00401614 8B8704000000    mov eax, dword ptr [edi+00000004]
:0040161A 89C3            mov ebx, eax
:0040161C 8B8708000000    mov eax, dword ptr [edi+00000008]
:00401622 39C3            cmp ebx, eax
:00401624 0F8D2E000000    jnl 00401658
:0040162A 8B8704000000    mov eax, dword ptr [edi+00000004]
:00401630 E8EBFBFFFF      call 00401220
:00401635 B886164000      mov eax, 00401686
:0040163A E8A1FBFFFF      call 004011E0
:0040163F 8B8704000000    mov eax, dword ptr [edi+00000004]
:00401645 89C3            mov ebx, eax
:00401647 B801000000      mov eax, 00000001
:0040164C 01C3            add ebx, eax
:0040164E 89D8            mov eax, ebx
:00401650 898704000000    mov dword ptr [edi+00000004], eax
:00401656 EBBC            jmp 00401614
```

```
* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00401624(C)
```

1

```
:00401658 B888164000      mov eax, 00401688
:0040165D E87EFBFFFF      call 004011E0
:00401662 E8A9FBFFFF      call 00401210
:00401667 E91CFDFFFF      jmp 00401388
:0040166C 0000000000000000 BYTE 10 DUP(0)
:00401676 0000000000000000 BYTE 10 DUP(0)
```

```
:00401680 00                                     BYTE 0
```

```
:00401681 2A00          sub al, byte ptr [eax]
:00401683 0020          add byte ptr [eax], ah
:00401685 0020          add byte ptr [eax], ah
:00401687 00000000000000000000000000000000  BYTE 10 DUP(0)
```

08

```
var K;
procedure P;
  procedure coma;
  begin
    write (',');
    K := K + 1;
    call P
  end;
begin
  if K < 10 then
    begin
      write (K);
      call coma;
    end
end;

begin
  K := 1;
  call P;
  writeln (10)
end.
```



```

000003d0    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
000003e0    60 89 c6 30 c0 02 06 74    08 46 e8 e1 fe ff ff eb    `..0...t.F.....
000003f0    f2 61 90 c3 00 00 00 00    00 00 00 00 00 00 00 00    .a.....
00000400    04 30 e8 c9 fe ff ff c3    00 00 00 00 00 00 00 00    .0.....
00000410    b0 0d e8 b9 fe ff ff b0    0a e8 b2 fe ff ff c3 00    .....
00000420    3d 00 00 00 80 75 4e b0    2d e8 a2 fe ff ff b0 02    =....uN.-.....
00000430    e8 cb ff ff ff b0 01 e8    c4 ff ff ff b0 04 e8 bd    .....
00000440    ff ff ff b0 07 e8 b6 ff    ff ff b0 04 e8 af ff ff    .....
00000450    ff b0 08 e8 a8 ff ff ff    b0 03 e8 a1 ff ff ff b0    .....
00000460    06 e8 9a ff ff ff b0 04    e8 93 ff ff ff b0 08 e8    .....
00000470    8c ff ff ff c3 3d 00 00    00 00 7d 0b 50 b0 2d e8    .....=....}.P.-.
00000480    4c fe ff ff 58 f7 d8 3d    0a 00 00 00 0f 8c ef 00    L...X..=.....
00000490    00 00 3d 64 00 00 00 0f    8c d1 00 00 00 3d e8 03    ..=d.....=.
000004a0    00 00 0f 8c b3 00 00 00    3d 10 27 00 00 0f 8c 95    .....=.'.....
000004b0    00 00 00 3d a0 86 01 00    7c 7b 3d 40 42 0f 00 7c    ...=....|{@B..|
000004c0    61 3d 80 96 98 00 7c 47    3d 00 e1 f5 05 7c 2d 3d    a=....|G=....|=
000004d0    00 ca 9a 3b 7c 13 ba 00    00 00 00 bb 00 ca 9a 3b    ...;|.....;
000004e0    f7 fb 52 e8 18 ff ff ff    58 ba 00 00 00 00 bb 00    ..R....X.....
000004f0    e1 f5 05 f7 fb 52 e8 05    ff ff ff 58 ba 00 00 00    .....R....X....
00000500    00 bb 80 96 98 00 f7 fb    52 e8 f2 fe ff ff 58 ba    .....R....X.
00000510    00 00 00 00 bb 40 42 0f    00 f7 fb 52 e8 df fe ff    .....@B....R....
00000520    ff 58 ba 00 00 00 00 bb    a0 86 01 00 f7 fb 52 e8    .X.....R.
00000530    cc fe ff ff 58 ba 00 00    00 00 bb 10 27 00 00 f7    ....X.....'...
00000540    fb 52 e8 b9 fe ff ff 58    ba 00 00 00 00 bb e8 03    .R.....X.....
00000550    00 00 f7 fb 52 e8 a6 fe    ff ff 58 ba 00 00 00 00    ....R....X.....
00000560    bb 64 00 00 00 f7 fb 52    e8 93 fe ff ff 58 ba 00    .d.....R....X..
00000570    00 00 00 bb 0a 00 00 00    f7 fb 52 e8 80 fe ff ff    .....R.....
00000580    58 e8 7a fe ff ff c3 00    ff 15 00 10 40 00 00 00    X.z.....@...
00000590    b9 00 00 00 00 b3 03 51    53 e8 a2 fd ff ff 5b 59    .....QS.....[Y
000005a0    3c 0d 0f 84 34 01 00 00    3c 08 0f 84 94 00 00 00    <...4...<.....
000005b0    3c 2d 0f 84 09 01 00 00    3c 30 7c db 3c 39 7f d7    <-.....<0|<9..
000005c0    2c 30 80 fb 00 74 d0 80    fb 02 75 0c 81 f9 00 00    ,0...t....u....
000005d0    00 00 75 04 3c 00 74 bf    80 fb 03 75 0a 3c 00 75    ..u.<.t....u.<.u
000005e0    04 b3 00 eb 02 b3 01 81    f9 cc cc cc 0c 7f a8 81    .....
000005f0    f9 34 33 33 f3 7c a0 88    c7 b8 0a 00 00 00 f7 e9    .433.|.....
00000600    3d 08 00 00 80 74 11 3d    f8 ff ff 7f 75 13 80 ff    =....t.=....u...
00000610    07 7e 0e e9 7f ff ff ff    80 ff 08 0f 8f 76 ff ff    .~.....v...
00000620    ff b9 00 00 00 00 88 f9    80 fb 02 74 04 01 c1 eb    .....t....
00000630    03 29 c8 91 88 f8 51 53    e8 c3 fd ff ff 5b 59 e9    .)....QS.....[Y.
00000640    53 ff ff ff 80 fb 03 0f    84 4a ff ff ff 51 53 b0    S.....J...QS.
00000650    08 e8 7a fc ff ff b0 20    e8 73 fc ff ff b0 08 e8    ..z....s.....
00000660    6c fc ff ff 5b 59 80 fb    00 75 07 b3 03 e9 25 ff    l...[Y...u....%.
00000670    ff ff 80 fb 02 75 0f 81    f9 00 00 00 00 75 07 b3    .....u.....u..
00000680    03 e9 11 ff ff ff 89 c8    b9 0a 00 00 00 ba 00 00    .....
00000690    00 00 3d 00 00 00 00 7d    08 f7 d8 f7 f9 f7 d8 eb    ..=....}).....
000006a0    02 f7 f9 89 c1 81 f9 00    00 00 00 0f 85 e6 fe ff    .....
000006b0    ff 80 fb 02 0f 84 dd fe    ff ff b3 03 e9 d6 fe ff    .....
000006c0    ff 80 fb 03 0f 85 cd fe    ff ff b0 2d 51 53 e8 fd    .....-QS..
000006d0    fb ff ff 5b 59 b3 02 e9    bb fe ff ff 80 fb 03 0f    ...[Y.....
000006e0    84 b2 fe ff ff 80 fb 02    75 0c 81 f9 00 00 00 00    .....u.....
000006f0    0f 84 a1 fe ff ff 51 e8    14 fd ff ff 59 89 c8 c3    .....Q.....Y...
00000700    bf 80 15 40 00 e9 52 00    00 00 e9 27 00 00 00 b8    ...@..R....'.....
00000710    84 15 40 00 e8 c7 fc ff    ff 8b 87 00 00 00 00 89    ..@.....
00000720    c3 b8 01 00 00 00 01 c3    89 d8 89 87 00 00 00 00    .....
00000730    e8 01 00 00 00 c3 8b 87    00 00 00 00 89 c3 b8 0a    .....
00000740    00 00 00 39 c3 0f 8d 10    00 00 00 8b 87 00 00 00    ...9.....
00000750    00 e8 ca fc ff ff e8 b4    ff ff ff c3 b8 01 00 00    .....
00000760    00 89 87 00 00 00 00 e8    ca ff ff ff b8 0a 00 00    .....
00000770    00 e8 aa fc ff ff e8 95    fc ff ff e9 08 fe ff ff    .....
00000780    00 00 00 00 2c 00 00 00    00 00 00 00 00 00 00 00    ....,.....
00000790    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
000007a0    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....

```

000007b0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000007c0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000007d0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000007e0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000007f0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

desensamblado

```

:00401500 BF80154000      mov edi, 00401580
:00401505 E952000000      jmp 0040155C
:0040150A E927000000      jmp 00401536

```

* Referenced by a CALL at Address:

|:00401556

```

|
:0040150F B884154000      mov eax, 00401584
:00401514 E8C7FCFFFF      call 004011E0
:00401519 8B8700000000    mov eax, dword ptr [edi+00000000]
:0040151F 89C3            mov ebx, eax
:00401521 B801000000      mov eax, 00000001
:00401526 01C3            add ebx, eax
:00401528 89D8            mov eax, ebx
:0040152A 898700000000    mov dword ptr [edi+00000000], eax
:00401530 E801000000      call 00401536
:00401535 C3            ret

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:0040150A(U)

```

|
:00401536 8B8700000000    mov eax, dword ptr [edi+00000000]
:0040153C 89C3            mov ebx, eax
:0040153E B80A000000      mov eax, 0000000A
:00401543 39C3            cmp ebx, eax
:00401545 0F8D10000000    jnl 0040155B
:0040154B 8B8700000000    mov eax, dword ptr [edi+00000000]
:00401551 E8CAFCFFFF      call 00401220
:00401556 E8B4FFFFF      call 0040150F

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401545(C)

```

|
:0040155B C3            ret

```

* Referenced by a (U)nconditional or (C)onditional Jump at Address:

|:00401505(U)

```

|
:0040155C B801000000      mov eax, 00000001
:00401561 898700000000    mov dword ptr [edi+00000000], eax
:00401567 E8CAFFFFF      call 00401536
:0040156C B80A000000      mov eax, 0000000A

```

```
:00401571 E8AAFCFFFF      call 00401220
:00401576 E895FCFFFF      call 00401210
:0040157B E908FEFFFF      jmp 00401388
:00401580 00000000      BYTE 4 DUP(0)

:00401584 2C00      sub al, 00
:00401586 00000000000000000000      BYTE 10 DUP(0)
```

09

.


```

000003d0    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
000003e0    60 89 c6 30 c0 02 06 74    08 46 e8 e1 fe ff ff eb    `..0...t.F.....
000003f0    f2 61 90 c3 00 00 00 00    00 00 00 00 00 00 00 00    .a.....
00000400    04 30 e8 c9 fe ff ff c3    00 00 00 00 00 00 00 00    .0.....
00000410    b0 0d e8 b9 fe ff ff b0    0a e8 b2 fe ff ff c3 00    .....
00000420    3d 00 00 00 80 75 4e b0    2d e8 a2 fe ff ff b0 02    =....uN.-.....
00000430    e8 cb ff ff ff b0 01 e8    c4 ff ff ff b0 04 e8 bd    .....
00000440    ff ff ff b0 07 e8 b6 ff    ff ff b0 04 e8 af ff ff    .....
00000450    ff b0 08 e8 a8 ff ff ff    b0 03 e8 a1 ff ff ff b0    .....
00000460    06 e8 9a ff ff ff b0 04    e8 93 ff ff ff b0 08 e8    .....
00000470    8c ff ff ff c3 3d 00 00    00 00 7d 0b 50 b0 2d e8    .....=....}.P.-.
00000480    4c fe ff ff 58 f7 d8 3d    0a 00 00 00 0f 8c ef 00    L...X..=.....
00000490    00 00 3d 64 00 00 00 0f    8c d1 00 00 00 3d e8 03    ..=d.....=.
000004a0    00 00 0f 8c b3 00 00 00    3d 10 27 00 00 0f 8c 95    .....='.....
000004b0    00 00 00 3d a0 86 01 00    7c 7b 3d 40 42 0f 00 7c    ...=....|{@B..|
000004c0    61 3d 80 96 98 00 7c 47    3d 00 e1 f5 05 7c 2d 3d    a=....|G=....|=
000004d0    00 ca 9a 3b 7c 13 ba 00    00 00 00 bb 00 ca 9a 3b    ...;|.....;
000004e0    f7 fb 52 e8 18 ff ff ff    58 ba 00 00 00 00 bb 00    ..R....X.....
000004f0    e1 f5 05 f7 fb 52 e8 05    ff ff ff 58 ba 00 00 00    .....R....X....
00000500    00 bb 80 96 98 00 f7 fb    52 e8 f2 fe ff ff 58 ba    .....R....X..
00000510    00 00 00 00 bb 40 42 0f    00 f7 fb 52 e8 df fe ff    .....@B....R....
00000520    ff 58 ba 00 00 00 00 bb    a0 86 01 00 f7 fb 52 e8    .X.....R..
00000530    cc fe ff ff 58 ba 00 00    00 00 bb 10 27 00 00 f7    ....X.....'...
00000540    fb 52 e8 b9 fe ff ff 58    ba 00 00 00 00 bb e8 03    .R....X.....
00000550    00 00 f7 fb 52 e8 a6 fe    ff ff 58 ba 00 00 00 00    ....R....X.....
00000560    bb 64 00 00 00 f7 fb 52    e8 93 fe ff ff 58 ba 00    .d....R....X..
00000570    00 00 00 bb 0a 00 00 00    f7 fb 52 e8 80 fe ff ff    .....R.....
00000580    58 e8 7a fe ff ff c3 00    ff 15 00 10 40 00 00 00    X,z.....@...
00000590    b9 00 00 00 00 b3 03 51    53 e8 a2 fd ff ff 5b 59    .....QS.....[Y
000005a0    3c 0d 0f 84 34 01 00 00    3c 08 0f 84 94 00 00 00    <...4...<.....
000005b0    3c 2d 0f 84 09 01 00 00    3c 30 7c db 3c 39 7f d7    <-.....<0|.<9..
000005c0    2c 30 80 fb 00 74 d0 80    fb 02 75 0c 81 f9 00 00    ,0...t....u....
000005d0    00 00 75 04 3c 00 74 bf    80 fb 03 75 0a 3c 00 75    ..u.<t....u.<u
000005e0    04 b3 00 eb 02 b3 01 81    f9 cc cc cc 0c 7f a8 81    .....
000005f0    f9 34 33 33 f3 7c a0 88    c7 b8 0a 00 00 00 f7 e9    .433.|.....
00000600    3d 08 00 00 80 74 11 3d    f8 ff ff 7f 75 13 80 ff    =....t.=...u...
00000610    07 7e 0e e9 7f ff ff ff    80 ff 08 0f 8f 76 ff ff    .~.....v...
00000620    ff b9 00 00 00 00 88 f9    80 fb 02 74 04 01 c1 eb    .....t....
00000630    03 29 c8 91 88 f8 51 53    e8 c3 fd ff ff 5b 59 e9    .)....QS.....[Y
00000640    53 ff ff ff 80 fb 03 0f    84 4a ff ff ff 51 53 b0    S.....J...QS.
00000650    08 e8 7a fc ff ff b0 20    e8 73 fc ff ff b0 08 e8    ..z....s.....
00000660    6c fc ff ff 5b 59 80 fb    00 75 07 b3 03 e9 25 ff    l...[Y...u...%.
00000670    ff ff 80 fb 02 75 0f 81    f9 00 00 00 00 75 07 b3    .....u.....u..
00000680    03 e9 11 ff ff ff 89 c8    b9 0a 00 00 00 ba 00 00    .....
00000690    00 00 3d 00 00 00 00 7d    08 f7 d8 f7 f9 f7 d8 eb    ..=....}.....
000006a0    02 f7 f9 89 c1 81 f9 00    00 00 00 0f 85 e6 fe ff    .....
000006b0    ff 80 fb 02 0f 84 dd fe    ff ff b3 03 e9 d6 fe ff    .....
000006c0    ff 80 fb 03 0f 85 cd fe    ff ff b0 2d 51 53 e8 fd    .....-QS..
000006d0    fb ff ff 5b 59 b3 02 e9    bb fe ff ff 80 fb 03 0f    ...[Y.....
000006e0    84 b2 fe ff ff 80 fb 02    75 0c 81 f9 00 00 00 00    .....u.....
000006f0    0f 84 a1 fe ff ff 51 e8    14 fd ff ff 59 89 c8 c3    .....Q.....Y...
00000700    bf 0a 15 40 00 e9 7e fe    ff ff 00 00 00 00 00 00    ...@...~.....
00000710    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
00000720    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
00000730    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
00000740    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
00000750    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
00000760    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
00000770    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
00000780    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
00000790    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....
000007a0    00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    .....

```

000007b0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000007c0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000007d0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000007e0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
000007f0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

Desensamblado:

:00401500 BF0A154000
:00401505 E97EFEFFFF

mov edi, 0040150A
jmp 00401388

3. Programas Incorrectos

00

```
var X, Y;

procedure INICIAR;
  const Y = 2;
  procedure ASIGNAR;
    X = Y;
  call ASIGNAR;

begin
  write ('NUM=') readln (Y);
  call INICIAR;
  writeln ("NUM*2=", Y*X)
end.
```

Intentos de compilación

Orden	Mensaje de error	Correccion
1	Expected := found =	Se reemplazo := por =
2	Expected END found READLN	Se agrego el caracter ';' entre write ('NUM=') y readln (Y);
3	Undeclared indentifier NUM	Se reemplazaron las comillas dobles por comillas simples
4	N/A	

01

```
VAR BASE, EXPO, RESU;

PROCEDUR POT;
IF EXPO > 0 THEN
  BEGIN
    RESU = RESU * BASE;
    EXPO := EXPO - 1;
    CALL POT
  END;

BEGIN
  WRITE ('BASE: '); READLN(BASE);
  WRITE ('EXPONENTE: '); READLN(EXPO)
  RESU := 1;
  CALL POT;
  IF EXPO < 0  RESU := 0;
  WRITELN ('RESULTADO: ', RESU);
  WRITELN
END.
```

Intentos de compilación

Orden	Mensaje de error	Correccion
1	Undeclared indentifier PROCEDUR	Se reemplazo el identificador mal escrito "PROCEDUR" por "PROCEDURE"
2	Expected := found =	Se reemplazo := por =
3	Expected END found IDENT	Se agrego el ';' faltante al final de la linea 13
4	Expected THEN found IDENT	Se agrego el THEN para cerrar la condicion del IF

02

```

var X, Y, Z;

procedure MULTIPLICAR;
var A, B, A;
begin
    A := X;
    B := Y;
    Z := 0;
    if X < 0 then A := -A;
    if Y ( 0 then B := -B;
    while B > 0 then
        begin
            if odd B then Z:= Z + A;
            A := A * 2;
            B := B / 2
        end;
    if X < 0 then Z:= -Z;
    if Y < 0 then Z:= -Z
end;

Begin
    write ('X: '); readLn X;
    write ('Y: '); readLn (Y);
    MULTIPLICAR;
    writeLn ('X*Y=', Z);
end.

```

Intentos de compilación

Orden	Mensaje de error	Correccion
1	Duplicated declaration: variable A	Se elimino la declaracion duplicada de A
2	Expected comparator found (Se reemplazo (por <
3	Expected DO found THEN	Se reemplazo THEN con DO
4	Expected (found IDENT	Se encerro el parametro de readLn entre parentesis
5	*Bad l-value: proc	Se puso la palabra CALL antes de MULTIPLICAR
6	N/A	

*Pero, ¿Porque “Bad l-value: proc”, un error semantico si niquiera es valido sintacticamente (asignacion incompleta)? Esto es asi porque al procesar el ident secuencialmente, lo asume como una asignacion de variable, pasa al analizador semantico el primer parametro que es el nombre del identificador, este lo valida y como el tipo de identificador es un procedure en este caso tira el error semantico antes de que hubiese chance intentar leer el proximo token

Si en lugar de “MULTIPLICAR” se hubiera puesto el nombre de una variable valida, entonces ahi habria aparecido el error sintactico (“Expected := found ;”)

03

```

var DO, X, Y, Q, R;

procedure DIVIDIR;
var V W;
begin
  Q := 0;
  R := X; if R < 0 then R := -R;
  W := Y; if W < 0 then W := -W;
  V := Y; if V < 0 then V := -V;
  while W <= R do W := W * 2;
  while W > V do
    begin
      Q := Q * 2; W := W / 2;
      if W <= R then
        begin
          R := R - W; Q := Q + 1
        end
      end;
    if X < 0 then R := -R;
    if X < 0 then Q := -Q;
    if Y < 0 then Q := -Q;
  end;

procedure OTRO;
  procedure DIVIDIR;
  begin
    Q := X / Y; R := X - Y * Q
  end;
call DIVIDIR;

procedure SALIDA;
begin
  write ('Cociente: ', Q); writeln;
  write ('Resto: ', R); writeln;
end;

begin
  write ('Dividendo: '); readln (X);
  write ('Divisor: '); readln (Y);
  writeln;
  if Y <> 0 then
    begin
      write ('Metodo 1'); writeln;
      call DIVIDIR;
    end
  end;
end;

```



```
        call SALIDA; writeln;  
        write ('Metodo 2'); writeln;  
        call OTRO;  
        call SALIDA;  
    end  
end  
end
```

Orden	Mensaje de error	Correccion
1	Expected IDENT found DO	Se elimino la palabra DO y la coma posterior, 'DO' no se puede usar como nombre de variable
2	Expected ; found IDENT	Se agrego una coma entre V y W
3	Expected . found END	Se reemplazo el ultimo "END" con un punto
4	N/A	

04

```
var X, Y Z;
```

```
procedure MCD
```

```
var F,G;
```

```
begin
```

```
  F := X; G := Y;
```

```
  while F <> G do
```

```
    begin
```

```
      if F < G then G := G - F;
```

```
      IF G < F then F := F - G
```

```
    end;
```

```
  Z:= F
```

```
end;
```

```
begin
```

```
  do write ('X: '); readln (X);
```

```
  if X > 0 then
```

```
    begin
```

```
      write ('Y: '); readln (Y);
```

```
      if Y > 0 then
```

```
        begin
```

```
          call MCD;
```

```
          writeln ('MCD: ', Z); writeln ()
```

```
        end
```

```
      end
```

```
end.
```

Orden	Mensaje de error	Correccion
1	Expected ; found IDENT	Se agrego una coma entre Y y Z
2	Expected ; found VAR	Se agrego un punto y coma despues del nombre del procedure
3	Expected END found DO	Se elimino la palabra DO
4	Expected IDENT, NUMBER or (found)	Se elimino el par de parentesis
5	N/A	

05

```
VAR R, N;
```

```
PROCEDURE INICIALIZAR;
```

```
CONST UNO = ;
```

```
R := -(-UNO);
```

```
PROCEDURE RAIZ;
```

```
BEGIN
```

```
    CALL N;
```

```
    WHILE R * R < N DO R := R + 1
```

```
END;
```

```
BEGIN
```

```
    WRITE ('N: '); READLN (N);
```

```
    WRITE ('RAIZ CUADRADA DE ', N, ': ');
```

```
    IF N < 0 THEN WRITE ('ERROR');
```

```
    IF N = 0 THEN WRITE (0);
```

```
    IF N > 0 THEN
```

```
        BEGIN
```

```
            CALL RAIZ;
```

```
            IF R* <>N THEN WRITE (R - 1, '..');
```

```
            WRITE (R);
```

```
        END;
```

```
    WRITELN
```

```
END.
```

Orden	Mensaje de error	Correccion
1	Expected NUMBER found ;	Se agrego el 1 antes del punto y coma y despues del =
2	Cannot call symbol of type _var	Se reemplazo N con INICIALIZAR
3	Expected IDENT, NUMBER or (found <>	Se agrego la R faltante
4	N/A	

06

CONST K1=1, K2=2, K3=3, K4=4, K5=5, K6=6, K7=7, K8=8, K9=9, K10=10,
 K11=11, K12=12, K13=13, K14=14, K15=15, K16=16, K17=17, K18=18,
 K19=19, K20=20,
 K21=21, K22=22, K23=23, K24=24, K25=25, K26=26, K27=27, K28=28,
 K29=29, K30=30,
 K31=31, K32=32, K33=33, K34=34, K35=35, K36=36, K37=37, K38=38,
 K39=39, K40=40,
 K41=41, K42=42, K43=43, K44=44, K45=45, K46=46, K47=47, K48=48,
 K49=49, K50=50,
 K51=51, K52=52, K53=53, K54=54, K55=55, K56=56, K57=57, K58=58,
 K59=59, K60=60,
 K61=61, K62=62, K63=63, K64=64, K65=65, K66=66, K67=67, K68=68,
 K69=69, K70=70,
 K71=71, K72=72, K73=73, K74=74, K75=75, K76=76, K77=77, K78=78,
 K79=79, K80=80,
 K81=81, K82=82, K83=83, K84=84, K85=85, K86=86, K87=87, K88=88,
 K89=89, K90=90,
 K91=91, K92=92, K93=93, K94=94, K95=95, K96=96, K97=97, K98=98,
 K99=99, K100=100,
 K101=101, K102=102, K103=103, K104=104, K105=105, K106=106,
 K107=107, K108=108, K109=109, K110=110,
 K111=111, K112=112, K113=113, K114=114, K115=115, K116=116,
 K117=117, K118=118, K119=119, K120=120,
 K121=121, K122=122, K123=123, K124=124, K125=125, K126=126,
 K127=127, K128=128, K129=129, K130=130,
 K131=131, K132=132, K133=133, K134=134, K135=135, K136=136,
 K137=137, K138=138, K139=139, K140=140,
 K141=141, K142=142, K143=143, K144=144, K145=145, K146=146,
 K147=147, K148=148, K149=149, K150=150,
 K151=151, K152=152, K153=153, K154=154, K155=155, K156=156,
 K157=157, K158=158, K159=159, K160=160,
 K161=161, K162=162, K163=163, K164=164, K165=165, K166=166,
 K167=167, K168=168, K169=169, K170=170,
 K171=171, K172=172, K173=173, K174=174, K175=175, K176=176,
 K177=177, K178=178, K179=179, K180=180,
 K181=181, K182=182, K183=183, K184=184, K185=185, K186=186,
 K187=187, K188=188, K189=189, K190=190,
 K191=191, K192=192, K193=193, K194=194, K195=195;

var IMPORTE, BILLETE, VUELTO, PESOS, CENTAVOS, VUELTOPESOS,
 VUELTOCENTAVOS,

IMPORTEOK, CENTAVOSOK, PESOSOK, BILLETEOK, VUELTOOK,
 DEMASIADOLARGO123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890,

```

V1, V2, V3, V4, V5, V6, V7, V8, V9, V10,
V11, V12, V13, V14, V15, V16, V17, V18, V19, V20,
V21, V22, V23, V24, V25, V26, V27, V28, V29, V30,
V31, V32, V33, V34, V35, V36, V37, V38, V39, V40,
V41, V42, V43, V44, V45, V46, V47, V48, V49, V50,
V51, V52, V53, V54, V55, V56, V57, V58, V59, V60,
V61, V62, V63, V64, V65, V66, V67, V68, V69, V70,
V71, V72, V73, V74, V75, V76, V77, V78, V79, V80,
V81, V82, V83, V84, V85, V86, V87, V88, V89, V90,
V91, V92, V93, V94, V95, V96, V97, V98, V99, V100,
V101, V102, V103, V104, V105, V106, V107, V108, V109, V110,
V111, V112, V113, V114, V115, V116, V117, V118, V119, V120,
V121, V122, V123, V124, V125;

```

```

procedure ERROR;
begin
    writeln ('VALOR FUERA DE RANGO!')
end;

```

```

begin
    WRITELN ('*****');
    writeln ('VUELTO PARA IMPORTES PAGADOS CON UN BILLETE');
    WRITELN ('*****');
    writeln;
    IMPORTEOK := -1;
    while IMPORTEOK <> 0 do
        begin
            writeln ('IMPORTE (min $0.01 y max $100.00)');
            CENTAVOSOK := -1;
            while CENTAVOSOK <> 0 do
                begin
                    write ('CENTAVOS: '); readLn (CENTAVOS);
                    CENTAVOSOK := 0;
                    if CENTAVOS < 0 then CENTAVOSOK := -1;
                    if CENTAVOS > 99 then CENTAVOSOK := -1;
                    if CENTAVOSOK <> 0 then call ERROR
                end;
            PESOSOK := -1;
            while PESOSOK <> 0 do
                begin
                    write ('PESOS: '); readLn (PESOS);
                    PESOSOK := 0;
                    if PESOS < 0 then PESOSOK := -1;
                    if PESOS > 100 then PESOSOK := -1;
                    if PESOSOK <> 0 then call ERROR
                end;
            end;

```



```

write ('IMPORTE: $', PESOS, '.');
if CENTAVOS < 10 then write ('0');
writeln (CENTAVOS);
IMPORTE := PESOS * 100 + CENTAVOS;
IMPORTEOK := 0;
if IMPORTE < 1 then IMPORTEOK := -1;
if IMPORTE > 10000 then IMPORTEOK := -1;
if IMPORTEOK <> 0 then call ERROR
end;

VUELTOOK := -1;
while VUELTOOK <> 0 do
begin
  BILLETEOK := -1;
  while BILLETEOK <> 0 do
begin
  write ('BILLETE (min $2 y max $100): $'); readLn
(BILLETE);
  BILLETEOK := -1;
  if BILLETE = 2 then BILLETEOK := 0;
  if BILLETE = 5 then BILLETEOK := 0;
  if BILLETE = 10 then BILLETEOK := 0;
  if BILLETE = 20 then BILLETEOK := 0;
  if BILLETE = 50 then BILLETEOK := 0;
  if BILLETE = 100 then BILLETEOK := 0;
  if BILLETEOK <> 0 then writeln ('BILLETE INEXISTENTE!')
end;
VUELTO := BILLETE * 100 - IMPORTE;
VUELTOOK := 0;
if VUELTO < 0 then
begin
  VUELTOOK := -1;
  writeln ('BILLETE INSUFICIENTE PARA EL PAGO!')
end
end;

VUELTOPESOS := VUELTO / 100;
VUELTOCENTAVOS := VUELTO - VUELTOPESOS * 100;

write ('Su vuelto: $', VUELTOPESOS, '.');
if VUELTOCENTAVOS < 10 then write ('0');
writeln (VUELTOCENTAVOS);

writeln;

if VUELTO >= 5000 then

```

```
begin
  writeLn ('1 billete de $50');
  VUELTO := VUELTO - 5000
end;

if VUELTO >= 2000 then
begin
  if VUELTO / 2000 = 1 then writeLn ('1 billete de $20');
  if VUELTO / 2000 <> 1 then writeLn ('2 billetes de $20');
  VUELTO := VUELTO - VUELTO / 2000 * 2000
end;

if VUELTO >= 1000 then
begin
  writeLn ('1 billete de $10');
  VUELTO := VUELTO - 1000
end;

if VUELTO >= 500 then
begin
  writeLn ('1 billete de $5');
  VUELTO := VUELTO - 500
end;

if VUELTO >= 200 then
begin
  if VUELTO / 200 = 1 then writeLn ('1 billete de $2');
  if VUELTO / 200 <> 1 then writeLn ('2 billetes de $2');
  VUELTO := VUELTO - VUELTO / 200 * 200
end;

if VUELTO >= 100 then
begin
  writeLn ('1 moneda de $1');
  VUELTO := VUELTO - 100
end;

if VUELTO >= 50 then
begin
  writeLn ('1 moneda de 50 centavos');
  VUELTO := VUELTO - 50
end;

if VUELTO >= 25 then
begin
  writeLn ('1 moneda de 25 centavos');
```

```

    VUELTO := VUELTO - 25
end;

if VUELTO >= 10 then
begin
    if VUELTO / 10 = 1 then writeLn ('1 moneda de 10 centavos');
    if VUELTO / 10 <> 1 then writeLn ('2 monedas de 10
centavos');
    VUELTO := VUELTO - VUELTO / 10 * 10
end;

if VUELTO >= 5 then
begin
    writeLn ('1 moneda de 5 centavos');
    VUELTO := VUELTO - 5
end;

if VUELTO > 1 then writeLn (VUELTO, ' monedas de 1 centavo');
if VUELTO = 1 then writeLn ('1 moneda de 1 centavo');

WRITELN
('*****
*****
*****
*****
*****
*****
*****');

WRITELN
('*****
*****
*****
*****');

WRITELN
('*****
*****
*****
*****')
end.

```

Orden	Mensaje de error	Correccion
1	Expected IDENT, NUMBER or (found EOF	Se eliminaron los ultimos tres WRITELN
2	N/A	

07

```

CONST N 20;

VAR A, B, C;

PROCEDURE TRI#NGULO;
VAR A, B;
BEGIN
  WRITELN;
  A := 15;
  WHILE A > 0 DO
    BEGIN
      B := 0;
      WHILE B < A DO
        BEGIN
          WRITE ('*')
          B := B + 1
        END;
      WRITELN;
      A := A - 1;
    END
  END;
END;

BEGIN
  A := 111111111111111111;
  WHILE A <= N DO
    BEGIN
      WRITE (A, ' ');
      A := A + TRIANGULO
    END;

    CALL TRIANGULO;

    B := -N;
    C := 0;
    WHILE B < C DO
      BEGIN
        WRITE (B, ' ');
        B := B + 1
      END;
      WRITELN;
    END.

```

Orden	Mensaje de error	Correccion
1	Expected = found NUMBER	Se agrego el = entre N y 20
2	Expected ; found IDENT	Se corrige el nombre del procedure "TRI#NGULO" a "TRIANGULO"
3	Expected END found IDENT	Se agrega el punto y coma faltante al final de la linea 15
4	Cannot read symbol of type proc	Se reemplaza "TRIANGULO" por 1 en la linea 28

Extra:

En la linea 24 se corrige la asignacion que deberia ser A:= 1

08

```
var K;  
procedure P;  
  procedure coma;  
  begin  
    write (',')  
    K := K + 1;  
    call P  
  end;  
begin  
  if K < 10 then  
    begin  
      write coma;  
      call K;  
    end  
  end;  
  
begin  
  K := 1;  
  call P;  
  writeln (10)
```

Orden	Mensaje de error	Correccion
1	Expected END found IDENT	Agregado punto y coma al final de la linea 5
2	Expected ; found IDENT	Se reemplaza el write con un call
3	Cannot call symbol of type _var	Se reemplaza el call con un write
4	Expected END found IDENT	Se agregan parentesis al parametro del write
5	Expected END found EOF	Se agrega END al final
6	Expected . found EOF	Se agrega un punto al final
7	N/A	

09

(NOTA: el programa consiste en una cadena de texto vacia)

Orden	Mensaje de error	Correccion
1	Expected . found EOF	Se agrega un punto al final
2	N/A	