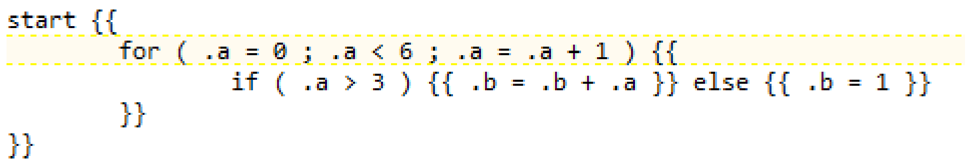
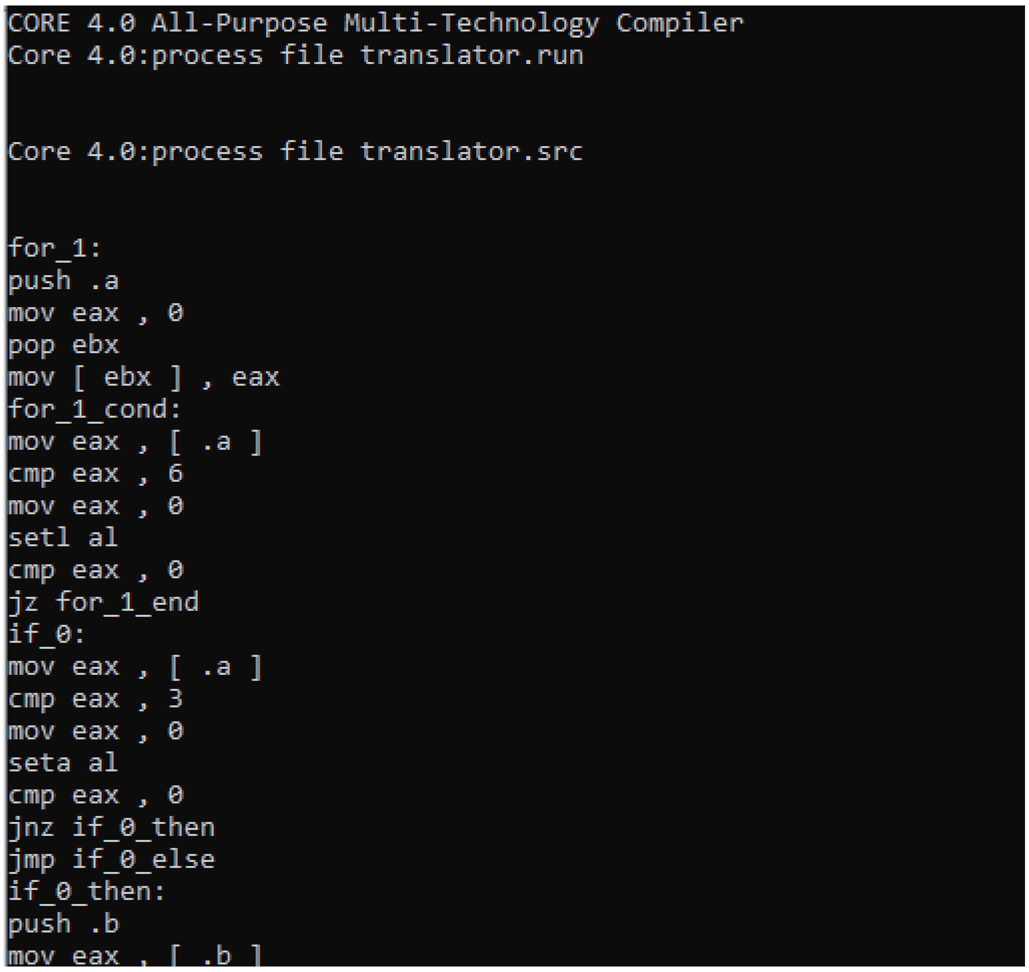
Core 4.0 Code to File

Today we don't do anything significant. So far we have tested the command line code, now we do it with the source files. Just write a source and process it with the appropriate command:



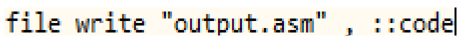
```
start {{
        for ( .a = 0 ; .a < 6 ; .a = .a + 1 ) {{
                if ( .a > 3 ) {{ .b = .b + .a }} else {{ .b = 1 }}
        }}
}}
```

And the code is translated:



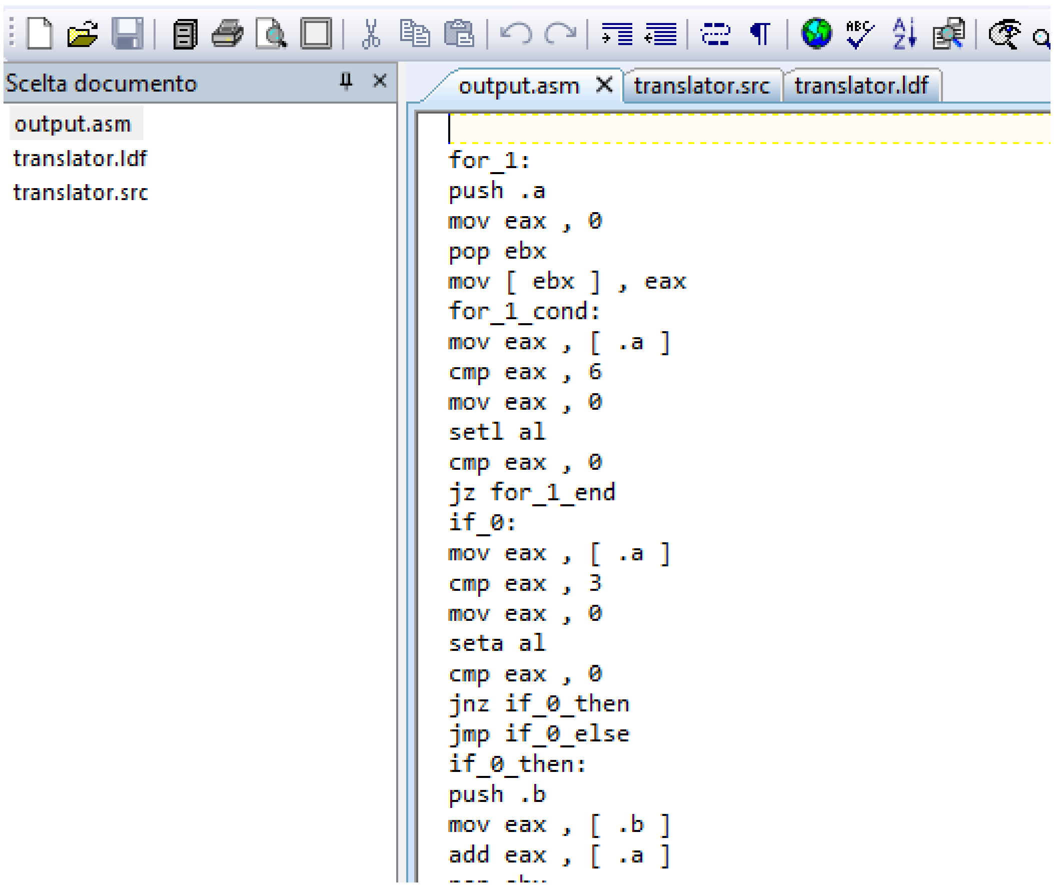
```
CORE 4.0 All-Purpose Multi-Technology Compiler
Core 4.0:process file translator.run


Core 4.0:process file translator.src


for_1:
push .a
mov eax , 0
pop ebx
mov [ ebx ] , eax
for_1_cond:
mov eax , [ .a ]
cmp eax , 6
mov eax , 0
setl al
cmp eax , 0
jz for_1_end
if_0:
mov eax , [ .a ]
cmp eax , 3
mov eax , 0
seta al
cmp eax , 0
jnz if_0_then
jmp if_0_else
if_0_then:
push .b
mov eax , [ .b ]
```

We can write it to a file, and paste it into this document, in courier. We will write:



```
file write "output.asm" , ::code
```

Here is the image file:



```
for_1:
push .a
mov eax , 0
pop ebx
mov [ ebx ] , eax
for_1_cond:
mov eax , [ .a ]
cmp eax , 6
mov eax , 0
setl al
cmp eax , 0
jz for_1_end
if_0:
mov eax , [ .a ]
cmp eax , 3
mov eax , 0
seta al
cmp eax , 0
jnz if_0_then
jmp if_0_else
if_0_then:
push .b
mov eax , [ .b ]
add eax , [ .a ]
```

And this is the entire code in courier:

```
for_1:
push .a
mov eax , 0
pop ebx
mov [ ebx ] , eax
for_1_cond:
mov eax , [ .a ]
cmp eax , 6
mov eax , 0
setl al
cmp eax , 0
jz for_1_end
if_0:
mov eax , [ .a ]
cmp eax , 3
mov eax , 0
seta al
cmp eax , 0
jnz if_0_then
jmp if_0_else
if_0_then:
```

```
push .b
mov eax , [ .b ]
add eax , [ .a ]
pop ebx
mov [ ebx ] , eax
jmp if_0_end
if_0_else:
push .b
mov eax , 1
pop ebx
mov [ ebx ] , eax
if_0_end:
for_1_onclycle:
push .a
mov eax , [ .a ]
add eax , 1
pop ebx
mov [ ebx ] , eax
jmp for_1_cond
for_1_end:
```

Thank you