```
.586
.model flat, stdcall
option casemap :none
.stack 4096

ExitProcess proto, dwExitCode :dword
GetStdHandle proto :dword
WriteConsoleA proto :dword, :dword, :dword, :dword, :dword
ReadConsoleA proto :dword, :dword, :dword, :dword, :dword

.data
    STD_INPUT_HANDLE equ -10
    STD_OUTPUT_HANDLE equ -11
    bufSize = 80
    inputHandle DWORD ?
    buffer db bufSize dup (?)
    bytes_read DWORD ?
    get_temp_input db "What temperature are you trying to convert? ",0
    final_output db "The temperature is ",0
    get_unit_input db "What unit are you wanting to convert from? ",0
    outputHandle DWORD ?
    bytes_written dd ?
    TemperatureOutput DWORD 0
    TemperatureInput dw 0
    TemperatureUnitInput byte 0
       TemperatureOutputString dd 0
    asciiBuf db 4 dup (0)
.code

main proc
    mov eax, 0
    mov ebx, 0
    mov ecx, 0
    mov edx, 0
    invoke GetStdHandle, STD_OUTPUT_HANDLE
    mov outputHandle, eax
    invoke GetStdHandle, STD_INPUT_HANDLE
    mov inputHandle, eax
    call GetTemperatureOutput
    call GetTemperatureInput;
    call GetUnitOutput
    call GetUnitInput
       mov TemperatureUnitInput, al
    call ComputeTemp
    mov TemperatureOutput, eax
;    call OutputComputed
    ret
main endp

GetTemperatureOutput PROC
    invoke GetStdHandle, STD_OUTPUT_HANDLE
    mov outputHandle, eax
    mov eax, LENGTHOF get_temp_input
    invoke WriteConsoleA, outputHandle, addr get_temp_input, eax, addr bytes_written, 0
    mov eax, 0
    mov eax, bytes_written
    ret
GetTemperatureOutput endp
```

```
GetTemperatureInput PROC
    invoke GetStdHandle, STD_INPUT_HANDLE
    mov inputHandle,eax
    invoke ReadConsoleA, inputHandle, addr buffer, bufSize, addr bytes_read, 0
    sub bytes_read, 2 ; -2 to remove cr,lf
       mov eax, 0
       mov ebx,0
       mov al, byte ptr buffer+[ebx]
       sub al,30h
       add TemperatureInput,ax
       getNext:
       inc bx
       cmp ebx,bytes_read
       jz finish
           mov ecx, 10
              mov ax, TemperatureInput
              mul ecx
              mov TemperatureInput, ax
              mov al, byte ptr buffer+[ebx]
              sub al,48
              add TemperatureInput,ax
              jmp getNext
       finish:
    ret
GetTemperatureInput endp

GetUnitOutput PROC
    invoke GetStdHandle, STD_OUTPUT_HANDLE
    mov outputHandle, eax
    mov eax, LENGTHOF get_unit_input
    invoke WriteConsoleA, outputHandle, addr get_unit_input, eax, addr bytes_written, 0
    mov eax, 0
    mov eax, bytes_written
    ret
GetUnitOutput endp

GetUnitInput PROC
    invoke GetStdHandle, STD_INPUT_HANDLE
    mov inputHandle,eax
    invoke ReadConsoleA, inputHandle, addr buffer, bufSize, addr bytes_read, 0
    sub bytes_read, 2
       mov eax, 0
    mov ebx, 0
       mov eax, bytes_read
       mov al, byte ptr buffer+[ebx]
       ret
GetUnitInput endp

ComputeTemp PROC
    mov al, TemperatureUnitInput
    cmp al, 70
    jz fahrenheit
    cmp al, 102
    jz fahrenheit
    mov eax, 0
    mov ax, TemperatureInput
    mov ecx, eax
```

```
    mov eax, 9
    imul ecx
       mov ecx, eax
    mov ecx, 5
       mov eax, eax
    idiv ecx
    add eax, 32
    ret
fahrenheit:
    mov eax, 0
    mov ax, TemperatureInput
    mov ecx, eax
       sub ecx, 32
    mov eax, 5
    imul ecx
       mov ecx, eax
    mov eax, 9
       mov ecx, ecx
    idiv eax
    ret
ComputeTemp endp


OutputComputed PROC
    invoke GetStdHandle, STD_OUTPUT_HANDLE
    mov outputHandle, eax
    mov eax, LENGTHOF final_output
    invoke WriteConsoleA, outputHandle, addr final_output, eax, addr bytes_written, 0
    mov eax, 0
    mov eax, bytes_written
    mov outputHandle, eax
       mov eax, LENGTHOF TemperatureOutput
    mov ebx, 10
       mov ecx, 0FFFFh
       mov edx, 0
       mov TemperatureOutput, 0
    mov TemperatureOutputString, 0
       mov eax, 0
       mov edi, 10000h
       mov ax, TemperatureInput
       mov eax, eax
       mov eax, edx
       mov edx, 0
       div ebx
       add eax, TemperatureOutputString
       add eax, 48
       add TemperatureOutputString, eax
       mov bytes_written, 0
parseStart:
       cmp TemperatureInput, 0
       jz endOfParse
           inc bytes_written
              mov eax, 0
              mov edx, 0
              mov ax, TemperatureInput
              mov eax, eax
              mov edx, 0
              div ebx
```

```
                add eax, 48
                mul edi
                add TemperatureOutputString, eax
                mov edx, eax
                mov TemperatureInput, ax
                jmp parseStart
endOfParse:
       invoke GetStdHandle, STD_INPUT_HANDLE
    mov eax, LENGTHOF TemperatureOutputString
    invoke WriteConsoleA, outputHandle, addr TemperatureOutputString, eax, addr
bytes_written, 0
    mov eax, 0
    mov eax, bytes_written
    mov outputHandle, eax
    ret
OutputComputed endp

end main
```