

```
; Attributes: bp-based frame

; int __cdecl main(int argc, const char **argv, const char **envp)
public main
main proc near

counts= byte ptr -22h
argc= dword ptr 8
argv= dword ptr 0Ch
envp= dword ptr 10h

lea    ecx, [esp+4]
and    esp, 0FFFFFFF0h
push   dword ptr [ecx-4]
push   ebp
mov    ebp, esp
push   ebx
push   ecx
sub    esp, 20h
mov    ebx, ecx
cmp    dword ptr [ebx], 1
jg     short loc_80486B5
```

The diagram illustrates the control flow from the main procedure to a usage message and then to a welcome message. A red arrow points from the 'jg' instruction in the main procedure to a green box containing the usage message code. A blue arrow points from the end of the usage message code to a green box containing the welcome message code.

```
mov    eax, [ebx+4]
mov    eax, [eax]
sub    esp, 8
push   eax
push   offset aUsageSInput_fi ; "Usage: %s input_file\n"
call   _printf
add    esp, 10h
sub    esp, 0Ch
push   1
call   _exit
```

```
loc_80486B5:
sub    esp, 4
push   1Ah
push   0
lea    eax, [ebp+counts]
push   eax
call   _memset
add    esp, 10h
sub    esp, 0Ch
push   offset aWelcomeToFrequ ; "Welcome to frequency analyzer"
call   _puts
add    esp, 10h
mov    eax, [ebx+4]
add    eax, 4
mov    eax, [eax]
sub    esp, 8
lea    edx, [ebp+counts]
push   edx
push   eax
call   analyze_input
add    esp, 10h
mov    eax, 0
lea    esp, [ebp-8]
pop    ecx
pop    ebx
pop    ebp
lea    esp, [ecx-4]
ret
main endp
```