

四個macro在我的電腦上定義在 `\usr\lib\gcc\x86_64-pc-linux-gnu\10.2.0\include\stdarg.h`

定義內容為

```
1 #define va_start(v, l)    __builtin_va_start(v, l)
2 #define va_end(v)        __builtin_va_end(v)
3 #define va_arg(v, l)      __builtin_va_arg(v, l)
4 #define va_copy(d, s)     __builtin_va_copy(d, s)
```

下載了gcc 10.1.0的source code 在裡面的 `gcc\builtins.c` 找到四個builtin function的定義

va\_start:

大致上是先檢查function 參數數量有沒有小於2，有的話就不能變成va\_list

再來檢查能不能被攤開(是不是沒有寫成不定長度...的樣子，或者是va\_start的參數數量對不對，是不是要的參數)

最後就攤開然後找到初始位置並回傳。

va\_end:

啥都不做，只是把丟進來的參數跟原本的va\_list解除關係(設定成NULL)

va\_arg:

由於這跟x86 calling convention 有很大的關係，基本上平台跟編譯器會有所不同，所以在builtins裡面找不到這個function的完整實做，不過以32bit來說他會把stack上的記憶體位置往後推你帶進來的型別的大小，便會找到下一個變數的位置了。

va\_copy:

找到複製到跟被複製的變數，如果沒有array type就直接複製，如果是的話轉成pointer在複製過去。