

1.

(1)因為 0 這個常數使用的頻率相當高，所以設定一個 register `$zero` 專門存放 0 這個常數可以提升效率。例如 `blt`, `bge`, `bgt`, `ble` 這四個 pseudo instructions，執行時都會用到 0 這個常數，若是沒有 `$zero`，則 MIPS compilers 每次都需要額外的指令來產生 0 這個常數。

(2)執行 `addi $zero, $zero, 5` 這個指令之後不會發生任何事，常數 5 不會被寫入，`$zero` 的值依然維持常數 0。

2.

callee 一開始先調整 stack pointer `$sp` 的值以取得需要的空間，再將需要保存的 argument 和 return address 等資料 push 到 stack，工作完成後，回復一開始保存在 stack 的資料，再調整 stack pointer `$sp` 的值將剛剛的資料 pop 出來以釋放空間，最後再返回 caller。

3.

我是非資電背景的跨考生，雖然對計算機組織和演算法的課程內容有基本的認識，但是在撰寫程式這方面我的程度就差強人意，如果這次作業助教沒有先提供範例程式作為參考，在缺乏提示的情況下我可能會卡在演算法，最後無法完成作業。儘管這次作業有滿足題目要求跑出結果，但是我寫的程式碼一定還有很多可以改進的地方。