Solution de la série 1

Exercice 2 :

```
org  100h        ; set location counter to 100h

jmp start
NOTES        DB      08, 06, 19, 11, 18 ; on donne 5 notes aléatoires pour vérifier l'exécution
                                        ; mais vous pouvez laisser les  indéfinis en mettant des ?
PLUS_G       DB      ?
start:

        MOV  CX, 5              ; compteur de boucle il se décrémente  automatiquement chaque itération

        MOV  BX, OFFSET NOTES           ; BX sera utilisé pour un adressage basé,
                                        ; il sert pour pointer  les données NOTES
        XOR    AL, AL                   ; Initialise AL à 0
        ENCORE: CMP  AL, [BX]   ; compare la note prochaine à la note la plus élevée
          JA      PROCHAIN              ; Sauter si AL est encore la note la plus élevée
        MOV  AL, [BX]           ; sinon AL retient la plus élevée
PROCHAIN:  INC    BX            ; pointe vers la prochaine note
        LOOP ENCORE             ; CX décrémente jusqu'à 0 pour sortir de la LOOP
        MOV   PLUS_G, AL ; sauvegarde de la note la plus élevée dans PLUS_G
;-------------------------------------------------------------------------------------------------------------------------
-------------------------

ret
```

Exercice 03 :

```
org  100h        ; set location counter to 100h

jmp start

 tab db 100,200,0,0,11, 75 dup( ?) ; on va déclarer une variable tab constituer de 80 octet
                                   ; les 5 premières valeurs ont été initialisées pour une éventuelle vérification


START:

mov cx, 5
xor al,al          ; pour initialiser al à zéro
lea bx,tab         ; cette instruction permet de charger du premier élement de tab
                   ; elle est équivalente à l'instruction mov bx, offset tab
encore:
mov dl, 0ffh
test [bx], dl
jnz nzero
inc al
nzero:
inc bx
loop encore
;on va stocker le résultat (nombres de zéro) dans la case mémoire qui correspond à l'adresse 300h
mov bx, 300h
mov [bx], al
ret
```