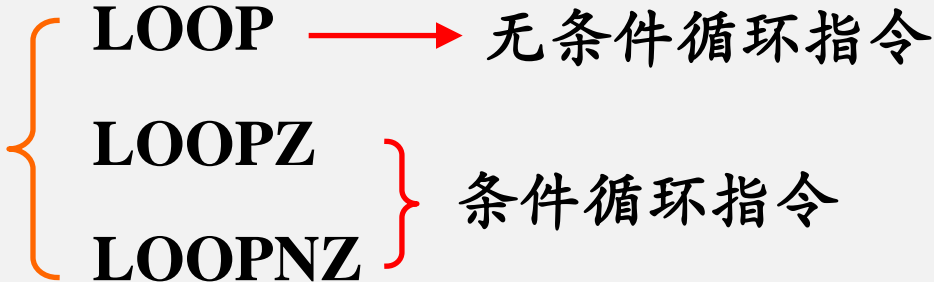


## 二、 循环控制指令

# 循环控制指令

- 循环范围：
  - 以当前IP为中心的-128~+127范围内循环。
- 循环次数由CX寄存器指定。
- 循环指令：
  - A diagram showing three instructions: LOOP, LOOPZ, and LOOPNZ. A large orange left curly bracket groups LOOPZ and LOOPNZ. A red arrow points from LOOP to the text '无条件循环指令' (unconditional loop instruction). A red right curly bracket groups LOOPZ and LOOPNZ, with the text '条件循环指令' (conditional loop instruction) to its right.
  - LOOP → 无条件循环指令
  - LOOPZ } 条件循环指令
  - LOOPNZ }

# 无条件循环指令

- 格式:

- `LOOP LABEL`

- 循环条件:

- `CX  $\neq$  0`

- 操作:

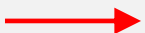
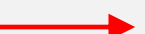
完全相当于 { `DEC CX`  
`JNZ 符号地址`

# 条件循环指令

- 功能:

- 先使CX-1, 再根据CX中的值及ZF值来决定是否继续循环

- 格式:

- **LOOPZ Label**  继续循环的条件: **CX≠0**, 且**ZF=1**
- **LOOPNZ Label**  继续循环的条件: **CX≠0**, 且**ZF=0**

# 例：

- 在以DATA为首地址的内存数据段中，存放有200个16位有符号数，试找出其中最大和最小的符号数，并分别放在MAX和MIN为首的内存单元中。
- 题目分析：
  - 先取数据块中第1个数，将其同时暂存于MAX和MIN中；
  - 循环读取其它数并分别与MAX和MIN中的数进行比较
    - 若大于则取代原MAX中的数；若小于MIN内容，则将新数放于MIN中。直到结束。

**注：**有符号数比较应采用JG和JL等用于符号数的条件转移指令

```
START: LEA SI, DATA
      MOV CX, 200
      CLD
      LODSW      → 读取2B到AX
      MOV MAX, AX
      MOV MIN, AX
      DEC CX
NEXT:  LODSW
      CMP AX, MAX
      JG LARGE
      CMP AX, MIN
      JL SMALL
      JMP GOON
LARGE: MOV MAX, AX
      JMP GOON ;
SMALL: MOV MIN, AX
GOON:  LOOP NEXT
      HLT
```

