

ساختارهای حلقه تکرار

در این صفحه دستورات حلقه و نحوه پیاده سازی ساختارهای حلقه های تکرار توضیح داده می شود.

[LOOP](#)
[LOOPE/LOOPZ](#)
[حلقه های تکرار دیگر](#)

LOOP

در زبان اسمبلی از دستور loop برای ساختن حلقه های شمارشی کاهشی با بدنه کوچک استفاده می شود. فرم کلی آن به صورت زیر است :

loop target

دستورالعمل loop ثبات CX را یک واحد کم می کند سپس اگر مقدار جدید CX مخالف با صفر باشد به آدرس target پرش می کند. اگر CX برابر با صفر باشد دستور بعدی اجرا می شود. ثبات CX به عنوان شمارنده عمل می کند و تعداد تکرار حلقه باید ابتدا در ثبات CX قرار بگیرد.

مثال. دستورات زیر مجموع اعداد ۱ تا ۱۰ را محاسبه و در ثبات AX ذخیره می کند.

```
mov AX, 0
mov CX, 10
lbl: add AX, CX
loop lbl
```

توجه کنید چون این دستور ابتدا از CX یک واحد کم می کند، اگر CX قبلاً حاوی صفر باشد بعد از کاهش برابر با مقدار ۶۵۵۳۵ می شود بنابراین حلقه ۶۵۵۳۶ بار تکرار خواهد شد.

دستورالعمل loop مشابه دستورات پرش کوتاه محدوده به ۱۲۸ بایت است و برای حلقه با بدنه کوچک مناسب است. برای حلقه های تکرار با بدنه بیشتر می توانید از دستورات پرش برای ساختن حلقه تکرار استفاده نمایید.

اگرچه از نام دستورالعمل loop اینطور برمی آید که حلقه تکرار ایجاد می کند اما به خاطر داشته باشید که تمام کاری که انجام می دهد این است که از ثبات CX یک واحد کم کرده اگر بعد از کاهش CX حاوی صفر نبود به آدرس مقصد پرش می کند.

دستورالعمل loop روی هیچکدام از ثبات ها تاثیر ندارد.

LOOPE/LOOPZ

دستورالعمل های loopz و loope مشابه دستور loop می باشند با این تفاوت که پرش در صورتی انجام می گیرد که CX مخالف با صفر و فلگ Z مساوی با یک باشد.

این دستورات برای استفاده بعد از دستور cmp مفید است.

مثال. دستورات زیر اولین عنصر غیرصفر را در آرایه ۱۵ تایی Array جستجو می کند. اگر کلیه عناصر آرایه صفر باشد بعد از اجرای حلقه به آدرس AllZero پرش می کند.

```
lea SI, Array
dec SI
mov cx, 15
Search:
inc SI
cmp byte ptr [SI], 0
loope Search
je AllZero
```

حلقه های تکرار دیگر

از دستورات هم پرش می توان برای ساختن حلقه های تکرار استفاده کرد. به مثال های زیر توجه کنید .

مثال. دستورات زیر مجموع اعداد ۱ تا ۱۰ را محاسبه و در ثبات AX ذخیره می کند .

```

mov AX, 0
mov CX, 1
For:
  cmp CX, 10
  jle Repeat
  jmp EndFor
Repeat:
  add AX, CX
  inc CX
  jmp For
EndFor:

```

مثال. دستورات زیر کاراکترهائی را از کاربر گرفته و در یک آرایه ذخیره می کند تا وقتی که کلید enter وارد شود .

```

lea SI, Array
Readkey:
  mov AH, 1
  int 21h      ;Get a character
  cmp AL, 13  ;Carriage return ASCII code
  je EndLoop
  mov [SI],AL ;Save input character in Array
  inc SI
  jmp Readkey
EndLoop:

```