

قالب اعداد:

در اینجا هر عدد به شکل مقابل نشان داده می شود.

$$111 \dots 1 = n$$

نعداد یک ها n می باشد.

قالب ورودی:

تابع ورودی $f(x, y)$ در نظر گرفته می شود. داریم:

$$\text{input}(x, y) = 11 \dots 1 \& 11 \dots 1$$

پس قالب ورودی می شود x تا 1 سپس یک علامت and و پس از آن y تا 1

قالب خروجی:

در صورتی که $x \geq y$ باشد قالب خروجی xy تا عدد 1 است در غیر این صورت خروجی صرفاً یک کاراکتر صفر است ('0')

$$TM = (Q, \Sigma, T, \delta, q_0, \blacksquare, F)$$

$$Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11}, q_{12}, q_{13}, q_{14}, q_{15}, q_{16}, q_{17}, q_{18}\}$$

$$\Sigma = \{1, \&\}$$

$$T = \{1, \&, =, X, \blacksquare\}$$

$$F = \{q_8, q_{18}\}$$

انتقال های ماشین تورینگ به همراه توضیحاتشان:

در تکه ی اول انتقالات کاری که انجام می شود مقایسه کردن x و y است با روشی که در هر مرحله با شروع از حالت q_0 اولین 1 سمت x را از سمت چپ تبدیل به X می کنیم سپس اولین 1 از سمت چپ y را تبدیل به X می کنیم این کار را انقدر ادامه می دهیم تا از قسمت x با از قسمت y دیگر 1 باقی نماند تا به X تبدیل شود.

$$\delta(q_0, 1) = (q_1, X, R)$$

$$\delta(q_1, 1) = (q_1, 1, R)$$

$$\delta(q_1, \&) = (q_2, \&, R)$$

$$\delta(q_2, X) = (q_2, X, R)$$

$$\delta(q_2, 1) = (q_3, X, L)$$

$$\delta(q_3, X) = (q_3, X, L)$$

$$\delta(q_3, \&) = (q_4, \&, L)$$

$$\delta(q_4, 1) = (q_4, 1, L)$$

$$\delta(q_4, X) = (q_0, X, R)$$

```

1&111
X&111
X&111
X&X11
X&X11
X&X11
X&X11
X&X11
X&X11
X&X11
X&X11

```

عکس بالا یک مثال از عملکرد این قسمت از برنامه است برای ورودی $x=1$ و $y=3$ که در هر مرحله اعضا داخل tape نشان داده می شود.

```

111&1
X11&1
X11&1
X11&1
X11&1
X11&X
X11&X
X11&X
X11&X
X11&X
XX1&X

```

عکس بالا یک مثال از عملکرد این قسمت از برنامه است برای ورودی $x=3$ و $y=1$ که در هر مرحله اعضا داخل tape نشان داده می شود.

توجه شود که در تمامی خروجی های نمایش داده شده اعضای داخل tape نمایش داده می شوند و اعضای نمایش داده نشده یا اعضا whitespace همان به معنی نبود کاراکتر است (همان ■)

پس از اجرای الگوریتم بالا دو حالت ممکن است که دیگر در سمت x عدد ۱ نماند یا در سمت y دیگر عدد ۱ نماند در این بخش حالت این که در قسمت x دیگر عدد ۱ نماند بررسی می شود اگر در سمت x عدد ۱ نماند پس باید در حالت ۰ و روی کاراکتر & باشیم حال با توجه به الگوریتم بالا و زوش تبدیل ۱ به X می دانیم $y \leq x$ می باشد باید بررسی کنیم که آیا $x = y$ است یا نه برای این کار در تکه تبدیلات زیر در رشته جلو می رویم اگر به عدد ۱ رسیدیم پس $y < x$ است و باید جواب ۰ شود پس در انتهای tape عدد ۰ قرار داده می شود سپس تمامی اعضای قبل آن حذف می شوند تا در نهایت به یک عدد ۰ تنها در tape برسیم

$$\delta(q_0, \&) = (q_5, \&, R)$$

$$\delta(q_5, X) = (q_5, X, R)$$

$$\delta(q_5, 1) = (q_6, 1, R)$$

$$\delta(q_6, 1) = (q_6, 1, R)$$

$$\delta(q_6, \blacksquare) = (q_7, 0, L)$$

$$\delta(q_7, 1) = (q_7, \blacksquare, L)$$

$$\delta(q_7, X) = (q_7, \blacksquare, L)$$

$$\delta(q_7, \&) = (q_7, \blacksquare, L)$$

$$\delta(q_7, \blacksquare) = (q_8, \blacksquare, L)$$

```

XX&XX1
XX&XX10
XX&XX 0
XX&X 0
XX& 0
XX 0
X 0
0

```

عکس بالا یک مثال از عملکرد این قسمت از برنامه است برای ورودی $x=2$ و $y=3$ که در هر مرحله اعضا داخل tape نسان داده می شود.

حال ممکن است حالت دیگر باشد یعنی $x \geq y$ باشد اگر این صورت باشد در آخر tape یک علامت مساوی می گذاریم تا بدانیم که از کجا به بعد می توانیم حاصل ضرب را قرار دهیم سپس روی tape پیمایش می کنیم تا اگر X دیدیم به جای آن ۱ بگذاریم و در نهایت زمانی که نشانگر به ابتدای tape اشاره می کند این قسمت از انتقالات تمام می شود

$$\delta(q_5, \blacksquare) = (q_9, =, L)$$

$$\delta(q_2, \blacksquare) = (q_9, =, L)$$

$$\delta(q_9, 1) = (q_9, 1, L)$$

$$\delta(q_9, X) = (q_9, 1, L)$$

$$\delta(q_9, \&) = (q_9, \&, L)$$

$$\delta(q_9, \blacksquare) = (q_{10}, \blacksquare, R)$$

```

XXX&XX
XXX&XX
XXX&XX
XXX&XX
XXX&XX=
XXX&X1=
XXX&11=
XXX&11=
XX1&11=
X11&11=
111&11=

```

مثال بالا برای $x=3$ و $y=2$ عملکرد انتقالات بالا را نشان می دهد که بعد از مقایسه کردن و یافتن رابطه ی بین x و y به سراغ گذاشتن علامت مساوی و ۱ کردن تمامی X ها که در مقایسه به X تبدیل شده اند می رویم

در قسمت بعدی زمانی که متوجه شدیم که $x \geq y$ است در هر مرحله به چپترین عنصر داخل tape نگاه می کنیم اگر یک بود آن را حذف می کنیم و سپس تمامی یک های بین & و = را کپی می کنیم و به آخر tape اضافه می کنیم با این کار در هر مرحله y تا یک اضافه می شود و چون x بار این کار را انجام می دهیم پس در نهایت xy تا ۱ جلوی مساوی داریم وقتی که دیگر سمت چپ tape عدد ۱ نباشد و به علامت and برسیم کار این قسمت از انتقالات تمام شده است.

$$\delta(q_{10}, 1) = (q_{11}, \blacksquare, R)$$

$$\delta(q_{11}.1) = (q_{11}.1.R)$$

$$\delta(q_{11}.\&) = (q_{12}.\&.R)$$

$$\delta(q_{12}.1) = (q_{13}.X.R)$$

$$\delta(q_{13}.1) = (q_{13}.1.R)$$

$$\delta(q_{13}.=) = (q_{14}.=.R)$$

$$\delta(q_{14}.1) = (q_{14}.1.R)$$

$$\delta(q_{14}.\blacksquare) = (q_{15}.1.L)$$

$$\delta(q_{15}.1) = (q_{15}.1.L)$$

$$\delta(q_{15}.=) = (q_{15}.=.L)$$

$$\delta(q_{15}.X) = (q_{12}.X.R)$$

$$\delta(q_{12}.=) = (q_{16}.=.L)$$

$$\delta(q_{16}.X) = (q_{16}.1.L)$$

$$\delta(q_{16}.\&) = (q_{16}.\&.L)$$

$$\delta(q_{16}.1) = (q_{16}.1.L)$$

$$\delta(q_{16}.\blacksquare) = (q_{10}.\blacksquare.R)$$

11&1=
 11&1=
 1&1=
 1&1=
 1&1=
 1&X=
 1&X=
 1&X=
 1&X=1
 1&X=1
 1&X=1
 1&X=1
 1&X=1
 1&1=1
 1&1=1
 1&1=1
 1&1=1
 1&1=1
 &1=1
 &1=1
 &X=1
 &X=1
 &X=1
 &X=1
 &X=11
 &X=11
 &X=11
 &X=11
 &X=11
 &X=11
 &1=11
 &1=11
 &1=11

در مثال $y=1$ و $x=2$ عملکرد انتقالات بالا بررسی شده است در هر مرحله محتوایات **tape** نمایش داده شده است روش کپی کردن عناصر y بعد علامت = همان روش کپی کردن داخل جزوه می باشد.

در نهایت تیکه انتقالات پایین عناصر قبل مساوی را پاک می کند

$$\delta(q_{10}, \&) = (q_{17}, \blacksquare, R)$$

$$\delta(q_{17}, 1) = (q_{17}, \blacksquare, R)$$

$$\delta(q_{17}, =) = (q_{18}, \blacksquare, R)$$

روش ورودی دادن به کد:

در ابتدا عدد x و سپس عدد y ورودی داده میشود با ورودی دادن این دو عدد رشته استاندارد که $11...1\&11...1$ می باشد که در ابتدا گفته شد تشکیل می شود و به ابتدای آن اشاره می کند سپس یک Boolean گرفته می شود که در صورت ۰ بودن تنها خروجی نمایش داده می شود در غیر این صورت محتوایات **tape** در هر مرحله بعد از انجام یک انتقال محتوایات **tape** خروجی داده میشود برای راحتتر بودن دیدن هر مرحله از انتقالات.

توضیحات بخش های مختلف کد کامنت شده است.

مثال هایی از ورودی و خروجی:

$x \quad y \quad \text{show_step}$

result

```
3 2 0
111111
```

```
4 2 0
11111111
```

```
4 3 0
111111111111
```

```
3 3 0
111111111
```

```
2 3 0
0
```

```
1 100 0
0
```