

در این پروژه باید حاصل دو برابر فاکتور سه برابر عددی را بدست آورم. برای حل این سوال با `turing` machine در ابتدا ، یک تابع به نام `multiply_turing` داریم که این تابع دو ورودی میگیرد و به عنوان خروجی ، حاصل ضرب آن ها را بر میگرداند.

در این تابع ابتدا به سه متغیر مقداردهی اولیه میکنیم و سپس مجموعه حالات با الفبای ورودی و حالتی که باید در `tape` نوشته شوند را به عنوان نوعی دیکشنری در پایتون تعریف میکنیم و سپس به متغیر `tape` کاراکتر  $\wedge$  اضافه میکنیم و به تعداد مقدار عدد اول به `tape` عدد 1 را `concat` میکنیم و یک \* میگذاریم و سپس به تعداد اندازه عدد دوم عدد یک را `concat` میکنیم و در این حالت نوآرمان را مقداردهی اولیه کرده ایم.

سپس با توجه به متغیر `pointer`، تابع `transition` را تعریف میکنیم و طبق دیکشنری که تعریف کردیم تا رسیدن به حالت نهایی رشته را به سمت جلو و عقب پیمایش میکنیم.

در حالت کلی که بخواهیم توضیح دهیم، به اندازه ای رشته را پیمایش میکنیم که به تعداد 1 های عدد سمت چپ، رشته ی سمت راست را در قسمت راست رشته ی اصلی کپی کرده باشیم.

در نهایت `final_tape` به دست میآید که در واقع یک عدد `unary` است و برای برگرداندن جواب به صورت `int` باید از تابع `count()` در پایتون استفاده کنیم و تعداد یک ها را بشماریم.

سپس پس از آن در تابع `main`، کاری که انجام میدهیم این است که ابتدا از ورودی یک عدد را میگیریم و سپس همان عدد را به همراه 3 به تابع `multiply_turing` میدهیم تا `3n` را بدست آوریم.

دقت کنیم که این اعداد در خود تابع `multiply_turing` به عدد `unary` تبدیل میشوند. حال برای یافتن فاکتوریل عدد `3n` باید از عدد 1 تا عدد `3n` را در هم ضرب کنیم. دقت کنیم که ضرب با ماشین تورینگ برای دو عدد `unary` عملیات بسیار وقت گیری است، زیرا باید به تعداد عدد اول ، عدد دوم را پیمایش کنیم و برای اعداد بزرگ تر از 30 که `unary` آنها بسیار طولانی است، این عمل بسیار وقتگیر خواهد بود و نیاز به قدرت پردازش بسیار زیادی خواهد داشت. برای مثال برای ضرب 20 در 700 باید 700 بار از روی عدد `unary` 20 عبور و مرور کنیم و اینکار از مرتبه ی `exponential` است.

در نهایت عدد به دست آمده را با عدد 2 به تابع `multiply_turing` میدهیم تا جواب نهایی بدست آید.

از آنجا که این عمل بسیار قدرت پردازشی زیادی میخواهد لذا نمیتوان برای `n` های بزرگ تکرار کرد و فقط تا `n=3` با کامپیوتر های عادی محاسبه کرد. لذا همان طور که در `screen shot` ها مشخص است به ازای

$n=1$  جواب 12، همچنین به ازای  $n=2$  پاسخ 1440، و همچنین به ازای  $n=3$ ، پس از مدت طولانی محاسبه عدد 725760 را برمیگرداند.