- A language $L$ is in $\text{DTIME}(f(n))$ if there is a deterministic TM $M$ with time complexity $f(n)$ such that $L(M) = L$.

- A language $L$ is in $\text{NTIME}(f(n))$ if there is a non-deterministic TM with time complexity $f(n)$.

- A language $L$ is in $\text{DSPACE}(f(n))$ if there is a deterministic TM with space complexity $f(n)$.

- A language $L$ is in $\text{NSPACE}(f(n))$ if there is a non-deterministic TM with space complexity $f(n)$. 
Complexity theory is about exploring general relations between the complexity classes $\text{DTIME}$, $\text{NTIME}$, $\text{DSPACE}$, $\text{NSPACE}$.

For example, if $L$ is in $\text{DTIME}(f(n))$, then $L$ is in $\text{DSPACE}(f(n))$.

Relation between $\text{DTIME}()$ and $\text{NTIME}()$:

If $L \in \text{DTIME}(f(n))$,

$$\Rightarrow L \in \text{NTIME}(f(n))$$

?? If $L \in \text{NTIME}(f(n))$, what does it imply for $\text{DTIME}()$?
NDTM simulated by a DTM

The cost of simulation is the number of nodes in the tree:

\[ O(d^{f(n)}) \]

\[ O(\log d) \cdot f(n) \]

For \( f(n) \) being a polynomial function, i.e. \( n^c \) for some fixed \( c \), we are confronted with the \( P = NP \) problem.
Claim: If $L$ is in the class $\text{DSPACE}(f(n))$ then $L$ is also in the class $\text{DSPACE}(c \cdot f(n))$ for any $c > 0$.

**Space Compression**

Read only input tape

Storage tape

We want to reduce the space by a factor $r = \frac{1}{c}$ (integer)

All $r$ tuples in the storage tape are compressed into a new symbol.

We must keep track of which of the $r$ cells the head is positioned on.

This can be done by increasing the number of states, so that once it is outside a window of $r$ cells, the new machine must actually move the head.
Claim: If \( L \in \text{DTime}(f(n)) \) then 
\[ L \in \text{DTime}(c \cdot f(n)) \]
if \( T(n) \in \omega(n) \)
\[ \left( \frac{T(n)}{n} \to \infty \right) \]

Time Compression

Will involve compressing space so that the new machine can read several symbols at once.

The head will definitely stay within the blue region.

To simulate \( r \) steps of the old machine, we somehow must precompute the \( r \)-fold composition of the transition function \( S \).
For the language $L = \{ w c w^r \}$
what is the time complexity
in a 1 tape TM.
(In a 2 tape TM, $L_{pal} \in \mathcal{N}$)
Consider all strings of the form
$\Sigma^{n/4} 0^{n/2} \Sigma^{n/4}$
which are palindromes.

$abc \overline{000000000} \overline{cba}$
Claim: For at least one input, the input is not unique. For any input with performance at or from the correct region and passing the correct sequence in the sequence from the following, we must have a unique input.
Observation: For any two distinct palindromes of the form
\[ w_1 \ 0^{n/2} \ w_1^R \]
\[ w_2 \ 0^{n/2} \ w_2^R \]
no two crossing sequences can be identical (in the window \(0^{n/2}\))

Then the TM can be fooled in \(i\) accepting \( w_1 \ 0^k \ w_2^R \ w_1 \neq w_2 \)
How many possible $w$, is $t$ length $n/4$?

For binary alphabet $2^n/4$.

So there must be at least $2^n/4$ distinct crossing sequences.

If the mean length crossing sequence (min over all possible inputs) for a fixed TM is $K$. Then the number of crossing sequences of length at most $K < n \sim |Q|^K$.

$|Q|^K > 2^n/4$.

$\Rightarrow K > \Omega \left( \frac{n}{\log |Q|} \right)$.