A language L is comp. enum. if it is either finite or

$\exists$ total computable $f : \mathbb{N} \xrightarrow[\text{onto}]{1\text{-}1} L$.

Theorem.

L is comp. enum. iff it is acceptable.

Proof ((Contd.)

L is acceptable $\Rightarrow$ L is c.e.

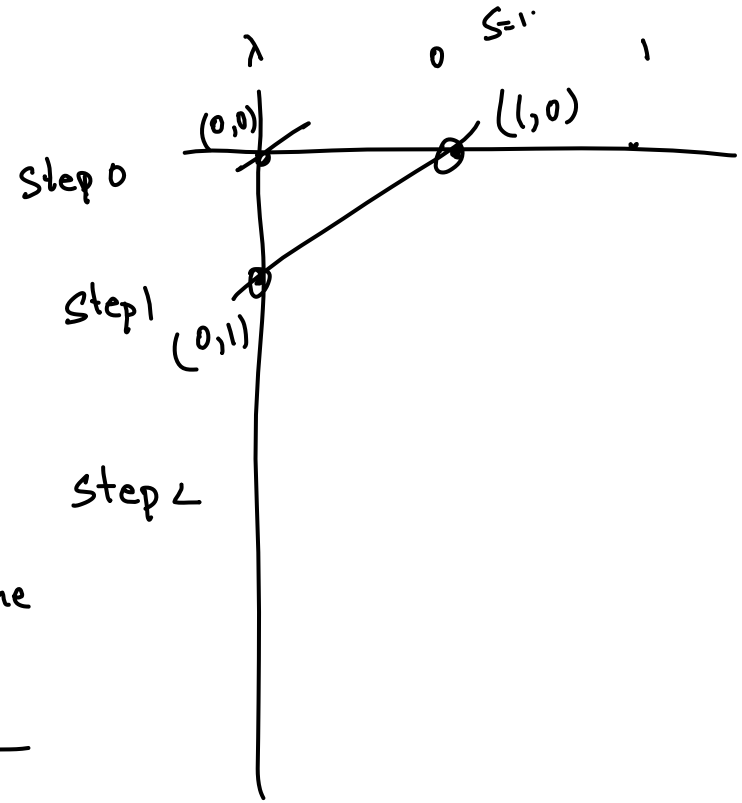
If L is finite then L is c.e. by definition, so assume L is infinite.

Since L is acceptable $\exists M$ such that $\forall x \in L$, $M(x)$ accepts & halts,

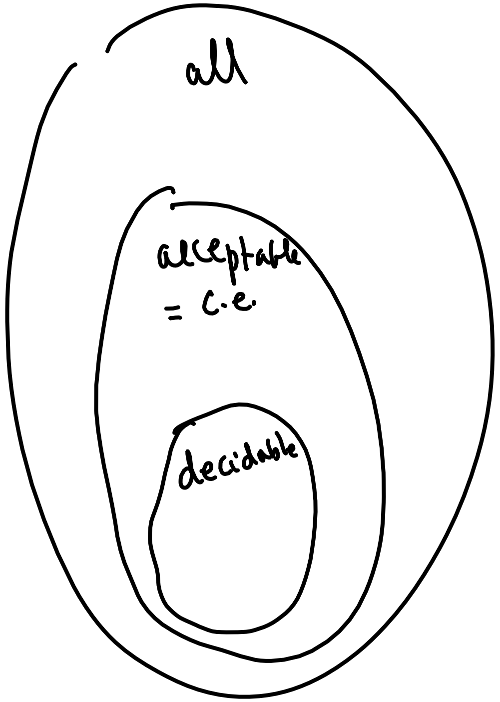
& $\forall x \notin L$ $M(x)$ does not accept (& may or may not halt.)

1. Input n
2. Output Set = $\emptyset$
3. for $s = 1$ to ∞ // "diagonal" index
4. for $i = 0$ to s
   - 4.1 run $M(s_j)$ for i steps.
   - 4.2 If s_j has been accepted,
   - 4.3 then OutputSet = OutputSet $\cup \{s_j\}$
   - 4.4 If $|\text{OutputSet}| \geq n$, then output the n^{th} string added.

$\hat{M}$



$\hat{M}(n)$ = n^{th} string added to the OutputSet in the dovetailing order.

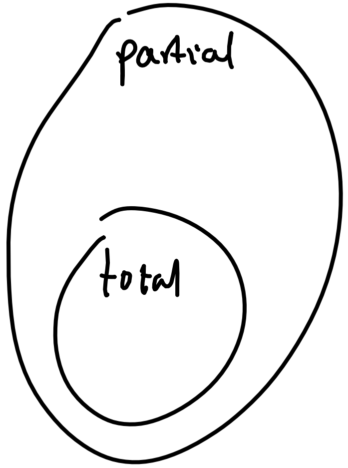


<u>H.W.</u> Show that L is decidable iff L is finite or $\exists$ total computable $f: \mathbb{N} \xrightarrow[\text{onto}]{1-1} L$ such that f enumerates L in increasing order.

acceptable = c.e.

decidable = c.e. in increasing order.

Now, we show that not every partial computable function is total.



Existence of "universal" partial computable functions

Theorem [Kleene's Normal Form Theorem]

There is a 3-argument partial computable function C and a 1-argument partial computable function U such that any 1-argument p.c. function $f_e : \Sigma^* \dashrightarrow \Sigma^*$ can be expressed as

$$f_e(n) = U\left[\underbrace{\mu z.}_{\text{"the smallest } z \text{ such that"}} \left[C(e, n, z) = 0\right]\right]$$