Since I have 15 mins:

Do part of the proof that the CFG G that I constructed from a PDA M last time actually generates the same strings that the PDA accepts.

If w is accepted by M, then we have some path of transitions

$$\underset{\text{state}}{(q_0},\ \underset{\text{left on string}}{w},\ \underset{\text{on stack}}{z)} \longrightarrow \cdots\cdots \longrightarrow (q_f, \lambda, \lambda)$$

This path of transitions should correspond to a derivation in G

$$(q_0, z, q_f) \longrightarrow \cdots\cdots \longrightarrow w$$

We show that single transitions in M correspond to steps in the derivation in G.

Accepting: aabb

~~$(q_0, aab$~~

	state	rem string	stack	derivation
	q_0	aabb	$\cancel{Z}$	(q_0, Z, q_2)
1	q_0	abb	$\cancel{A}Z$	$a(q_0, A, q_1)(q_1, Z, q_2)$
2	q_0	bb	AAZ	$a\cancel{b}a(q_0, A, q_1)(q_1, A, q_1)(q_1, Z, q_2)$
3	q_1	b	AZ	$aab(q_1, A, q_1)(q_1, Z, q_2)$
4	q_1	λ	$\cancel{A}$	$aabb(q_1, Z, q_2)$
5	q_2	λ	λ	aabb

we are at q_1 when we get to just Z on stack,

at q_1 when we get to A Z on stack

Again I have to find a loop invariant relating the derivation and my path through the PDA.

The letters at the beginning of the derived string tell us the letters consumed as input so far

The variables at the end ~~are~~ correspond to the ~~variables~~ letters on the stack - they are (q, X, q'), where q is the state we're in having eaten the previous letter off the stack, X is the stack letter, q' is the state we're in after eating X from stack.

<u>i.e</u>: The current step in the derivation looks like $w V_1 \cdots V_k$ where

1) w is the string consumed so far
2) $V_i = (q_{i-1}, A_i, q_i)$ where A_i is the i-th letter from the top of the stack.

q_i is the state we are in after taking A_i off the stack. (maybe this should be before taking A_{i+1} off the stack).

If my transition puts nothing on the stack, this corresponds to using the production $(q_1, A_1, q_2) \longrightarrow \ell$, and this gives us the string

$$\longrightarrow w\ell\,(q_1, A_2, q_2) \cdots\cdots (\qquad\qquad)$$