CPSC 433: Lab Exercise
And-Tree-Based Search

Assume we are going to write a program sort recyclables. You have at your disposal a robot (R) that can move around an NxN grid that contains recyclable objects in the categories of glass (G), paper (P), or Plastic (PL) as well as bins for each (IGI, IPI, and IPLI). For example:

IGI	IPLI						
		G	R				
				G			P
			G	G	G	G	G
			G	PL	PL	PL	
			IPI			P	
			G			P	

The robot can do 6 actions only: *up, down, left, right, grab, drop* with the obvious interpretations. The robot must be in the same cell as an object to grab it, and when it drops it, the object remains in the same cell. The robot can then only move back onto the cell from which it arrived. The robot can move onto the same cell as an object (garbage or bin), but may only return to the cell from which it came. Thus, in the example above, the plans "left, right" and "left, grab, drop, right" are legal, but the plans "left, left" and "left, grab, drop, left" are not. The robot may only carry one object at at time, and may not move/carry a bin, only a garbage object. Your job is come up with a plan, which is a series of actions, that will clear all garbage on the grid into the correct bins.

a) Design an and-tree based model (A=(S,T)) by defining the types <u>and</u> definitions of S, and T, together with any auxiliary definitions as are used in search paradigm. **Hint:** You should view this problem as finding a *plan*, so Div is centred around dividing up a *plan*, <u>not</u> dividing up the grid or the garbage types, etc. To do this, you may want to use higher-level plans, which you must define in terms of actions (and maybe other plans), such as "moveto(cell)" (some series of up, down, left, right that will move the robot from the current location to the specified cell) or "disposeof(cell)" (move the object in *cell* to the appropriate bin). (Your answers may be in text; they do not necessarily have to be in formal notation, although full marks will only be given if the answer includes formal definitions.)

b) Define your search process P=(A, Env, K) by defining the types <u>and</u> definitions of Env, and K, together with any auxiliary definitions as are used in the search paradigm.

c) Define G and s₀ for the following problem:

IGI	IPLI	IPI
		R
P	PL	G

and draw enough of the tree to demonstrate how your design works.