Research review: I will summarize three historical developments in classical planning; STRIPS, POP and GRAPHPLAN.

STRIPS: STRIPS (Stanford Research Institute Problem Solver) project was an attempt at solving problems faced by an autonomous robot called Shakey during the early 1970s. It led to many developments including the discovery of A*star and STRIPS planner(Fikes and Nilsson 1993). The major contribution of STRIPS to the planning field was a new formulation for describing the agent's actions(Fikes and Nilsson 1993). As we have also learned in the lecture videos by Peter Norvig, describing an agent's action in formal logic was very cumbersome. For example, for an action that painted one ball in a box that contained one hundred balls, we had to specify that colors of all the other balls did not change. STRIPS formulation made a simple assumption that an action changed only the aspects of the world specifically mentioned in actions' add and delete lists. This is also called "STRIPS assumption" or closed-world assumption. This was also the approach we used during this assignment. Another important contribution of STRIPS was an execution monitor(Fikes and Nilsson 1993). Execution monitor scanned the world after each action and checked whether the current state of the world is still suitable for the continued execution of the original plan. Although the preference was given to executing the original plan, the monitor also checked whether the next action really needs to executed because some of goal literals may serendipitously end up being true.

Partial order planning: During 1970s the popular approach to planning was linear planning. The goal was subdivided into sub goals, a plan was constructed for each sub goal. However this approach was shown to be insufficient through the discovery of Sussman anomaly(Sussman 1975).One solution to Sussman anomaly was to search through the space of plans instead of space of states through partial order planning (POP)(Sacerdoti 1975). One advantage of searching through the space of plans is that actions can be inserted anywhere along the plan not just in the end. The idea of POP (as apposed to total ordering) is to practice the least amount of commitment and only to record essential ordering decisions. In fact POP planer returns not just one plan but a set of plans, for example (Action1<Action2 and Action1<Action3) would be a POP plan because Action2 and Action3 can happen in any order. Ordering decisions of actions are made to prevent threats between actions. We say that A1 threatens A2->A3 order if A1 is between A2 and A3 and it negates either preconditions of A3 or effects of A2. In this case A1 either needs to placed before A2(demolition) or after A3(promotion). Regressive POP searches through the space of plans by adding new actions and preventing threats until all the goal literals are supported by links to null actions whose effects are initial state literals. Today POP planners are no longer competitive, used only for special purposes (Russell, Norvig et al. 2010).

GRAPHPLAN algorithm: GRAPHPLAN algorithm is based on planning graphs that are covered in the assignment, and was discovered by Blum and Furst (Blum and Furst 1997). Graphplan proceeds in two stages, the first stage is the expansion of the graph through adding actions(action layers), the resulting new literals(state layers), and the mutex relations. The second stage of graphplan algorithm is plan extraction. Plan extraction starts as soon all the goal literals are present in the final state level. This is a necessary but insufficient condition for a plan to exist. Graphplan starts at the last level of the graph by choosing a non-mutex set of actions for each of the goal literals). This constitutes a backtrack point, as there maybe more than one set available. Next we move into a lower level and recursively search for a plan for the set of preconditions of actions chosen previously. The base case for this recursion is level zero; if all the preconditions of chosen actions from the previous level are supported by non-conflicting actions we have a plan, otherwise we backtrack to a previous choice point until we ran out of options. If we have no solution we can keep expanding the graph plan until it levels off. Today Graphplan is still a competitive algorithm, but it is not clear whether it will scale up to more complex planning problems(Russell, Norvig et al. 2010).

References:

Blum, A. L. and M. L. Furst (1997). "Fast planning through planning graph analysis." Artificial Intelligence **90**(1-2): 281-300.

Fikes, R. E. and N. J. Nilsson (1993). "Strips, a Retrospective." Artificial Intelligence **59**(1-2): 227-232.

Russell, S. J., P. Norvig and E. Davis (2010). Artificial intelligence : a modern approach. Upper Saddle River, Prentice Hall.

Sacerdoti, E. D. (1975). The nonlinear nature of plans. Proceedings of the 4th international joint conference on Artificial intelligence - Volume 1. Tblisi, USSR, Morgan Kaufmann Publishers Inc.**:** 206-214.

Sussman, G. J. (1975). A computer model of skill acquisition. New York, American Elsevier Pub. Co.