335-05 Project 2 Asgwilanga Caverns – Big-O Analysis

For this program, there are two main functions being called: get_min_residue() and update().

get_min_residue() finds the minimum residue of the rooms in the blue causeway fulfilling the zero or max and ID limit rules. The function contains a nested loop iterating through N * M times, N being the number of possible IDs for the first ID and M being the number of possible IDs for the next ID. The function iterates through those IDs and calculates the last ID and checks if at least one of the integer is zero or max. The functions inside of the loops are O(1). The Big-O of get_min_residue() is O(N*M).

update() pushes the current roomid into an array, draws connections between rooms, draw rooms, finds the next room, and recursively calls itself if there is another room. Pushing onto an array is O(1). Draw connection is O(1). Calculating the residue for what room to draw is O(1) and drawing the room itself is also O(1). find_next_room() has a nested loop, S being the possible IDs being kept the same and T being the zero or max values of each room id. All the equality statements and switch statements in find_next_room() is O(1). Check limit is O(1). check_history() has a nested loop, L being the length of the history and K being the length of each room ID. check_history() is O(L*K). find_next_room() should be O(S*T*L*K). update() will recursively call itself if a room is found. The recursive loop is O(N). The Big-O of update() should be O(N*S*T*L*K).

The Big-O of the overall program should be O(N*S*T*L*K) because of the update function.