1. **Explain the heuristic that can be used to solve the problem? Justify your choice.**
2. We have created different combinations of packets having maximum weight 10 kg called Sets# at the start and that will remain same throughout the algorithm. 10 kg is the maximum weight robot can take at a time.  
   *Ex- Below are few possible sets.  
   Set-1: 1 kg+ 1 kg + 1 kg+ 1 kg + 1 kg + 5 kg  
   Set-2: 2 kg + 2 kg + 6 kg*

*Set-3: 2 kg + 4 kg + 4 kg  
………………………….*

*………………………….*

*………………………….*

As creating set is random, while running the code we have noticed that 13 or 14 such sets will be created. Let’s say 13 Sets are created for this run.

1. Robot will pick any random set from above 13 sets and start its commute to rooms.

*Ex-Let’s say Robot picks set-1.*

1. Algorithm will select any Random room.

*Ex-Let’s say Random room selected was of 5 kg capacity.*

1. Robot will place the packets in the room up to the room’s capacity.

*Ex-Robot will place 1+1+1+1+1 kg or 5 kg packet in the selected room.*

1. If the room selected in step 4 is full and robot still has some packets, robot will randomly select a new not previously selected room and place the remaining packets there.

*Ex-Room is full, so robot will select a new room let’s say-20 kg capacity, Robot will drop remaining 5 kg in this room.*

1. Robot will come to reception and randomly selects a new set of packets.

*Ex-Let’s say robot selects Set-3*

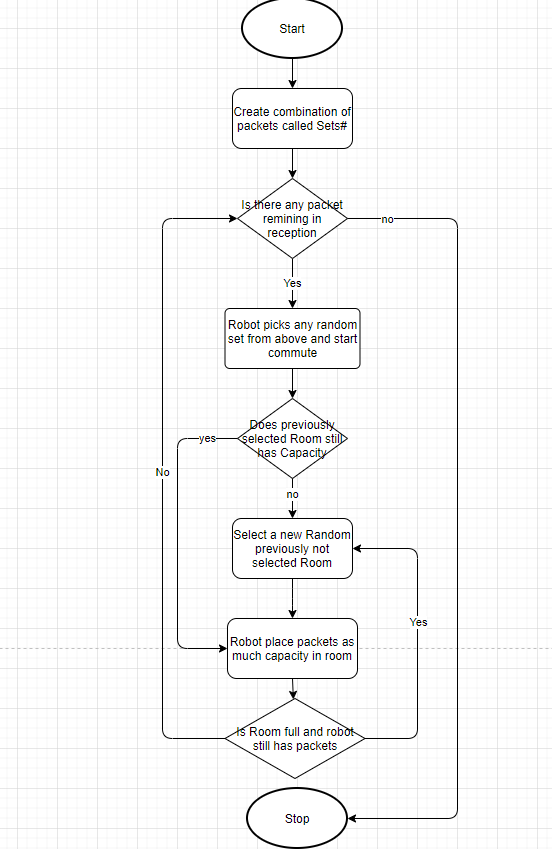
1. Robot will go to the same room of step-3 or step-4 which still has capacity to accommodate new packets. If all the rooms are fully occupied which is algorithms best case, robot will select a not previously selected new random room.

*Ex-Robot will drop all packets (10 kg weight) in 20 kg room.*

1. In this way robot has taken all the packets.

Now, we will be calculating below two points:

1. Space wasted: It is possible that few rooms are partially occupied and 1 kg, 2 kg fragmented space is still available.
2. Unselected Rooms: There is possibly that few rooms are fully empty.



1. **Explain the cost function associated with your search in reaching the goal.**

We have randomly grouped the packets in Sets# and total weight is fixed 10 kg per commute, therefore the number of commutes will always be same.

In this way we are saving the commute.

As number of commutes will remain same, we are ignoring it in cost function.

As we are filling rooms as much as possible and then moving to the next room, there are high chances that rooms are fully occupied.

We have considered room utilization as priority. Selecting fewer rooms and Optimizing space.

Also, for the number of rooms utilized and the space wastage, take the geometric mean.

**objective function calculation = ∑**

**sum of all utilized rooms as (remaining space/total room space)**

**- All (unutilized room space/total storage space of warehouse) of**

*Ex- Minimum of (5/25+ 2/20+……. – 5/151 – 6/151 …..)* **To be discussed**

**Running 100 times and select minimum as solution**

1. **Choose the correct algorithm suitable for this grid search.**

We have chosen Hill Climbing algorithm.

We have defined a cost function in such a way that minima of that cost function will give us the solution. Which means we need to do a local search on our objective function.

We have used hill climbing to search minima.

1. **Representation of the environment, fringe and the data structures used**  
   We know that there are different types of agents in AI. PEAS System is used to categorize similar agents together. The PEAS system delivers the performance measure with respect to the environment, actuators and sensors of the respective agent.

PEAS stand for Performance measure, Environment, Actuator, Sensor.

Performance Measure: Performance measure is the unit to define the success of an agent. Performance varies with agents based on their different precept.

Environment: Environment is the surrounding of an agent at every instant. It keeps changing with time if the agent is set in motion.

There are 5 major types of environments:

Fully Observable & Partially Observable

Episodic & Static

Static & Dynamic

Discrete & Continuous

Deterministic & Stochastic

Actuator: Actuator is a part of the agent that delivers the output of an action to the environment.

Sensor: Sensors are the receptive parts of an agent which takes in the input for the agent.

Example:

| **Agent** | **Performance Measure** | **Environment** | **Actuator** | **Sensor** |
| --- | --- | --- | --- | --- |
| Hospital Management System | Patient’s health, Admission process, Payment | Hospital, Doctors, Patients | Prescription, Diagnosis, Scan report | Symptoms, Patient’s response |
| Automated Car Drive | Comfortable trip, Safety, Maximum Distance | Roads, Traffic, Vehicles | Steering wheel, Accelerator, Brake, Mirror | Camera, GPS, Odometer |
| Subject Tutoring | Maximize scores, Improvement is students | Classroom, Desk, Chair, Board, Staff, Students | Smart displays, Corrections | Eyes, Ears, Notebooks |
| Packet Delivery System (Assignment example) | Space utilization  Number of Commutes  Time taken | Number of Rooms  Capacity of Rooms  Area over which rooms are spread  Accessibility of rooms  Conveyor belt  Bins | Algorithm  Jointed arm  hand | Camera  GPS  Sensor  Weighing sensor  Joint angle sensors |