You are given the following defined class ROOM with the only features you need for this problem.

```
class ROOM
feature {ANY}
status : STATUS          -- One of reserved, unreserved, occupied, repair
  guest : GUEST          -- Only for HOTEL void unless occupied or reserved.
end
```

Write require, ensure and class invariant assertions for the following methods of a class HOTEL that represents rooms and guests at a hotel. The minimum size of a hotel is 100 rooms. Write your assertions in as formal a mathematical notation as possible. Your assertions do not have to be executable.

```
class HOTEL  create make
feature

avail_rooms  : LIST[ROOMS]     -- List of all the rooms not under repair
                               -- available for use
repair_rooms : LIST[ROOMS]     -- List of all the rooms under repair
capacity     : INTEGER         -- Number of rooms in the hotel

make (size : INTEGER)
 -- Build a new hotel with size rooms where all rooms are available.
  require
  ensure


vacancy : BOOLEAN
 -- Returns true if and only if there is an unreserved room.
  require
  ensure


unreserved_check_in (guest : GUEST)
 -- The guest has not made a reservation.  Puts the guest into an unreserved room.
  require
  ensure


remove_room_for_repair (room : ROOM)
 -- Moves an unoccupied room from the available list to the repair list.
  require
  ensure


vacancy_count : INTEGER
 -- Returns the number of unreserved rooms.
  require
  ensure


invariant

end
```