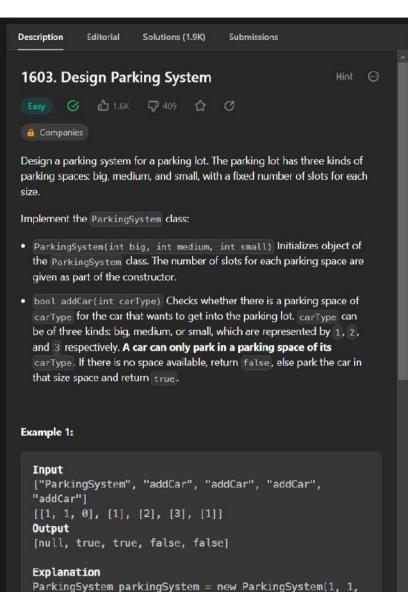


```
i Java ∨ | - Auto
  1 class Solution {
         public int divide(int dividend, int divisor) {
              if (dividend == 1<<31 && divisor == -1) return Integer.MAX VALUE;
             boolean sign = (dividend>=0) == (divisor >=0) ? true : false;
              dividend = Math.abs(dividend);
             divisor = Math.abs(divisor);
              int result =0;
              while(dividend - divisor >=0){
                  int count =0;
                  while(dividend - (divisor << 1 << count) >= 0 ){
                      count++:
                  result +=1 <<count;
                  dividend -= divisor <<count;</pre>
Testcase
Accepted Runtime: 0 ms
  • Case 1
             • Case 2
Input
  10
  3
Output
  3
Expected
                                                                                                             Submit
Console V
                                                                                                    Run
```



parkingSystem.addCar(1); // return true because there

asskipatustom addCar(3). (/ ratura trus bassusa

is 1 available slot for a big car

```
Java V - Auto
          Press Esc to exit full screen int[] space;
          public ParkingSystem(int big, int medium, int small) {
              space = new int[]{big, medium, small};
          public boolean addCar(int carType) {
              if(space[carType - 1] == 0){
              space[carType - 1]--;
              return true;
Testcase
         Result
Accepted Runtime: 0 ms
 Case 1
Input
  ["ParkingSystem","addCar","addCar","addCar"]
  [[1,1,0],[1],[2],[3],[1]]
Output
  [null, true, true, false, false]
Expected
  [null, true, true, false, false]
                                                                                                               Submit
Console ~
                                                                                                      Run
```