BY: Fadi Alahmad Alomar 120180049

Question 2:

0	1	2	3	4	5	6	7	8	9	10	11	12
-	27	-	-	-	10	19	-	-	-	36	-	-

Two collision happens with 10 one with 36 and one with 27 until it is put in the place 5

Question 3:

Because if we have three collisions and we delete the element that is in the middle and we search for the last element we will not find it even though it is there

Example:

H(x) = x mod 5

Numbers = 5,10,15

Result

0	1	2	3	4
5	10	15	-	-

Delete 10 and search for 15

0	1	2	3	4
5	-	15	-	-

```
The search for 15 will result in False

# Question 4:

FUNCTION FindFirstRepetition(arr):
    h <- hash table
    for i in arr:
        if i in h:
            return i
        else:
            h[i] <- 1
        end
    end
    return NO REPETING ELEMENT
end


# Question 6:

FUNCTION FindMostOnLine(arr):
    h <- hash table
    max <- -inf
    for i=0 ,i < arr.size:
        for j=i+1 , j<arr.size:
            x1 <- arr[i][0]
            y1 <- arr[i][1]
            x2 <- arr[j][0]
            y2 <- arr[j][1]
            m <- (y1-y2)/(x1-x2)
            c <- y1 - m*x1
            if (m,c) in h:
                h[(m,c)] <- h[(m,c)] + 1
            else:
                h[(m,c)] <- 1
            end
            if h[(m,c)] > max:
                max <- h[(m,c)]
            end
        end
    end
    return max
end


# Question 8:

FUNCTION FindIfConsecutive(arr):
    h <- hash table
    max <- - inf
```

```
    min <- inf
    for i in arr:
        if i in h:
            return False
        else:
            h[i] <- 1
        end
        if i > max:
            max <- i
        end
        if i < min:
            min <- i
       end
    end
    if max - min + 1 == arr.size:
        return True
    end
    return False
end
```