bool containsDuplicate(int\* nums, int numsSize){

    int HashTable[numsSize];

    int count[numsSize];

    //printf("numsSize:%d\n",numsSize);

    bool result=false;

    for(int i = 0 ; i < numsSize ; i ++ ){

        HashTable[i]=NULL;

        count[i]=0;

    }

    //printf("count:");

    for(int i=0 ;i < numsSize ; i++)

    {       int a = nums[i] ;

            if( a < 0 ){

                a=a\*-1;

            }

            a=a%numsSize;

            //printf("nums:%d:a=%d:",nums[i],a);

            if( nums[i] == 0){

                HashTable[0]=nums[i];

                count[0]++;

                if( count[0] >= 2){ return result=true;}

            }else if( a == 0 && nums[i] != 0){

                for(int j=1 ; j<numsSize ; j++){

                    if( HashTable[j]==nums[i] || HashTable[j] == NULL){

                        HashTable[j]=nums[i];

                        count[j]++;

                        if( count[j] >= 2){ return result=true;}

                        break;

                    }

                }

            }else if( HashTable[a] != NULL && HashTable[a] != nums[i]){

                //printf("\*\*%d,collison,%d: ",nums[i],a);

                for(int j=1 ; j<numsSize ; j++){

                    //printf("N=%d>>",nums[i]);

                    //printf("H[%d]=%d>>",j,HashTable[j]);

                    //printf("C[%d]=%d< ",j,count[j]);

                    if( HashTable[j]==nums[i] || HashTable[j] == NULL){

                        HashTable[j]=nums[i];

                        count[j]++;

                        if( count[j] >= 2){ return result=true;}

                        //printf("HashTable:%d >",HashTable[j]);

                        //printf("count:%d >",count[j]);

                        break;

                    }

                }

            }else{

                HashTable[a]=nums[i];

                count[a]++;

                if( count[a] >= 2){ return result=true;}

            }

            //printf("%d ",count[a]);

    }

    //printf("\n");

    //printf("check");

    return result;

}