int findMin(int\* nums, int numsSize){

    int r=numsSize-1;

    int l=0;

    int mid=(numsSize-1)/2;

    int right\_ver=nums[r];

    int left\_ver=nums[l];

    int mid\_ver=nums[mid];

    int ans;

    while(r > l){

        right\_ver=nums[r];

        left\_ver=nums[l];

        mid\_ver=nums[mid];

        //printf("left\_ver=%d mid\_ver=%d right\_ver=%d\n",left\_ver,mid\_ver,right\_ver);

        if( mid\_ver >= right\_ver && mid\_ver >= left\_ver){//[3,4,5,1,2]

            if( right\_ver > left\_ver){

                r=mid-1;

            }else{

                l=mid+1;

            }

            mid = (r+l)/2;

        }else if( mid\_ver <= right\_ver &&  mid\_ver >= left\_ver){//[1,2,3,4,5]

            if( right\_ver > left\_ver){

                r=mid-1;

            }else{

                l=mid+1;

            }

            mid = (r+l)/2;

        }else if( mid\_ver <= right\_ver &&  mid\_ver <= left\_ver){//[4,5,1,2,3]//[5,1,2,3,4]

                r=r-1;

                l=l+1;

        }

    }

    if( mid\_ver > right\_ver){

        ans=right\_ver;

        if(ans > left\_ver){

             ans=left\_ver;

        }

    }else{

        ans=mid\_ver;

        if(ans > left\_ver){

        ans=left\_ver;

        }

    }

return ans;

}