Tony Zhong, S3C7

#countEvens

public int countEvens(int[] nums) {

int count = 0;

for (int i = 0; i < nums.length; i++)

if (nums[i] % 2 == 0) count++;

return count;

}

#bigDiff

public int bigDiff(int[] nums) {

int max = nums[0];

int min = nums[0];

for (int i = 0; i < nums.length; i++) {

if (nums[i] > max) max = nums[i];

if (nums[i] <= min) min = nums[i];

}

return max - min;

}

#centeredAverage

public int centeredAverage(int[] nums) {

int max = nums[0];

int min = nums[0];

int sum = 0;

for (int i = 0; i < nums.length; i++) {

sum += nums[i];

if (nums[i] > max) max = nums[i];

if (nums[i] < min) min = nums[i];

}

return (sum - (max + min)) / (nums.length - 2);

}

#sum13

public int sum13(int[] nums) {

int total = 0;

for (int i = 0; i < nums.length; i++) {

if (nums[i] != 13) total += nums[i];

else if (i <= nums.length - 1) i++;

}

return total;

}

#sum67

public int sum67(int[] nums) {

int sum = 0;

for (int i = 0; i < nums.length; i++) {

if (nums[i] != 6) sum += nums[i];

else

while (nums[i] != 7) i++;

}

return sum;

}

#has22

public boolean has22(int[] nums) {

for (int i = 0; i <= nums.length - 2; i++)

if (nums[i] == 2 && nums[i + 1] == 2) return true;

return false;

}

#luck13public boolean lucky13(int[] nums) {

for (int i = 0; i < nums.length; i++)

if (nums[i] == 1 || nums[i] == 3) return false;

return true;

}

#sum28

public boolean sum28(int[] nums) {

int sum = 0;

for (int i = 0; i < nums.length; i++)

if (nums[i] == 2) sum += 2;

return sum == 8;

}

#more14

public boolean more14(int[] nums) {

int ones = 0;

int fours = 0;

for (int i = 0; i < nums.length; i++) {

if (nums[i] == 1) ones++;

if (nums[i] == 4) fours++;

}

return ones > fours;

}

#only14

public boolean only14(int[] nums) {

for (int i = 0; i < nums.length; i++)

if (nums[i] != 1 && nums[i] != 4) return false;

return true;

}