#include <limits.h>

int max(int a, int b) {

return (a > b) ? a : b;

}

int max3(int a, int b, int c) {

return max(max(a, b), c);

}

int maxCrossingSum(int\* nums, int l, int m, int h) {

int sum = 0;

int left\_sum = INT\_MIN;

for (int i = m; i >= l; i--) {

sum += nums[i];

if (sum > left\_sum)

left\_sum = sum;

}

sum = 0;

int right\_sum = INT\_MIN;

for (int i = m + 1; i <= h; i++) {

sum += nums[i];

if (sum > right\_sum)

right\_sum = sum;

}

return left\_sum + right\_sum;

}

int helper(int\* nums, int l, int h) {

if (l == h)

return nums[l];

int m = (l + h) / 2;

return max3(helper(nums, l, m),

helper(nums, m + 1, h),

maxCrossingSum(nums, l, m, h));

}

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

return helper(nums, 0, numsSize - 1);

}

