1. **eligible.m**

function ok = eligible(v,q)

ok = mean([v q]) >= 92 && v > 88 && q > 88;

end

1. **fare.m**

function cost = fare(miles,age)

cost = 2;

miles = round(miles);

if miles <= 10

cost = cost + max([0 (miles-1)]) \* 0.25;

else

cost = cost + 9 \* 0.25 + (miles-10) \* 0.1;

end

if age <= 18 || age >= 60

cost = cost \* 0.8;

end

end

1. **sort3.m**

function [a b c] = sort3(v)

if v(2) < v(1)

v = [v(2) v(1) v(3)];

end

if v(3) < v(2)

v = [v(1) v(3) v(2)]; % the last element is the max now

end

if v(2) < v(1)

v = [v(2) v(1) v(3)]; % the first two are in order as well

end

a = v(1);

b = v(2);

c = v(3);

end

1. **day\_diff.m**

function diff = day\_diff(m1,d1,m2,d2)

days = [31 28 31 30 31 30 31 31 30 31 30 31];

diff = -1;

if ~isscalar(m1) || m1 < 1 || m1 > 12 || m1 ~= floor(m1), return;

elseif ~isscalar(m2) || m2 < 1 || m2 > 12 || m2 ~= floor(m2), return;

elseif ~isscalar(d1) || d1 < 1 || d1 > days(m1) || d1 ~= floor(d1), return;

elseif ~isscalar(d2) || d2 < 1 || d2 > days(m2) || d2 ~= floor(d2), return;

end

if m2 < m1 || (m1 == m2 && d2 < d1) % make sure m1/d1 is the earlier date

tmp = m1; m1 = m2; m2 = tmp;

tmp = d1; d1 = d2; d2 = tmp;

end

diff = sum(days(m1:m2-1)) + d2 - d1;

end

1. **holiday.m**

function yes = holiday(month,day)

yes = ((month == 1 && day == 1) || (month == 7 && day == 4) || ...

(month == 12 && (day == 25 || day == 31)));

end

1. **poly\_val.m**

function p = poly\_val(c0,c,x)

c = c(:); % make sure c and x are column vectors

x = x(:);

if isempty(c)

p = c0;

else

p = c0 + sum(c' .\* x .^ (1:length(c)));

end

end

1. **exp\_average.m**

function res = exp\_average(x,b)

persistent avg;

persistent alpha;

if isempty(avg)

avg = x;

alpha = 0.1;

end

if nargin == 2

avg = x;

alpha = b;

end

avg = x \* alpha + avg \* (1-alpha);

res = avg;

end

1. **spherical\_mirror\_abber.m**

function mbd = spherical\_mirror\_aberr(fn,D)

deltax = 1e-2;

x = 0:deltax:D/2;

f = fn \* D;

theta = asin(x / 2 / f);

d = 2 \* f \* tan(2 \* theta) .\* (1 ./ cos(theta) - 1);

mbd = 8 \* deltax / (D^2) \* sum(x .\* d);

end