```
% homework18 Solutions

% Given values in homework
timeVector = 140:149;
co2Rate = [15.72 15.53 15.19 16.56 16.21 17.39 17.36 17.42 17.60
 17.75];
o2Rate = [15.49 16.16 15.35 15.13 14.20 14.23 14.29 12.74 14.74
 13.68];

% Using Simpson_Bechara
co2tot_bechara = Simpson_Bechara(timeVector,co2Rate);
o2tot_bechara = Simpson_Bechara(timeVector,o2Rate);

% Using MATLAB trapz()
co2tot_trapz = trapz(timeVector, co2Rate);
o2tot_trapz = trapz(timeVector, o2Rate);

% Output
fprintf('\n ------------------------ TOTAL CO2 EVOLUTION
 --------------------------');
fprintf('\n Total carbon dioxide evolution = %9.7f (evaluated by the
 trapezoidal rule)',co2tot_trapz);
fprintf('\n Total carbon dioxide evolution = %9.4f (evaluated by the
 simpson 1/3 rule)\n',co2tot_bechara);
fprintf('\n ------------------------- TOTAL O2 UPTAKE
 -------------------------');
fprintf('\n Total oxygen uptake = %9.4f (evaluated by the trapezoidal
 rule)',o2tot_trapz);
fprintf('\n Total oxygen uptake = %9.4f (evaluated by the simpson 1/3
 rule)\n\n',o2tot_bechara);
```

-1 if header is incorrect but the function runs

```
function [ I ] = Simpson_Bechara( x,y )
%SimpsonBechara Numerical evaluation of integral by Simpson's 1/3 Rule
%(composite)
%   I = Simpson_Bechara(x,y)
% Inputs
%   x = the vector of equally spaced independent variable
%   y = the vector of function values with respect to x
% Outputs:
%   I = the numerical integral calculated

% Error check that the inputs are the same length
if length(x) ~= length(y)
    error('x and y are not the same length');
end
```

+2, error check for same length vectors

```
% Error check that the x input is equally spaced
dx = diff(x);
if max(dx)-min(dx) ~= 0
    error('x is not equally spaced');
end
```

+2, error check for unequally spaced data

```
% Warn if trapezoidal rule is used on last interval
if mod(length(x),2) == 0
    warning('Odd number of intervals detected; Trapezoidal rule will
 be used for the last interval')
    n = length(x) - 1;
else
    n = length(x);
end

% Integration

h = dx(1);
y1 = y(2:2:n-1);
y2 = y(3:2:n-2);
I = (y(1) + 4*sum(y1) + 2*sum(y2) + y(n)) * h/3;

if n ~= length(x)
    I = I + (y(length(x)) + y(n)) * h/2;
end
end
```

+2, warning statements

```
Warning: Odd number of intervals detected; Trapezoidal rule
will be used for the last interval
Warning: Odd number of intervals detected; Trapezoidal rule
will be used for the last interval

 ------------------------- TOTAL CO2 EVOLUTION
 --------------------------
 Total carbon dioxide evolution = 149.9950000 (evaluated by the
 trapezoidal rule)
 Total carbon dioxide evolution =  150.4883 (evaluated by the simpson
 1/3 rule)
```

+1 correct value for trapezoidal rule, +1 correct value for simpson's rule

```
 ------------------------- TOTAL O2 UPTAKE -------------------------
 Total oxygen uptake =  131.4250 (evaluated by the trapezoidal rule)
 Total oxygen uptake =  131.1933 (evaluated by the simpson 1/3 rule)
```

+1 correct value for trapezoidal rule, +1 correct value for simpson's rule

Published with MATLAB® R2017a

+2, output statement or some comparison between the values