

---

```
% lab2_2024.m
% Please place lab2.m in your working directory
% Provide the print-out from running this function
% using 'publish lab2'
%
% T. Holton 7 Feb 2024
```

```
clear
test_lab2_2024;
```

## Real-time Convolution

Real-time convolution #1

```
x = [1 4 2 6 5];
h = [4 -1 3 -5 2];
test_lab2_2024;
test_lab2_2024(x, h);
```

```
% Real-time convolution convolution #2
test_lab2_2024(h, x);
```

```
% Real-time convolution #3
x = cos(2 * pi * (1:50000) / 16); % nice, big sequence
h = ones(1, 10);
tic;
test_lab2_2024(x, h);
t = toc;
disp(['The long convolution took ' num2str(t) ' secs'])
disp(' ')
```

```
Real-time convolution #1
    Your data are correct
```

```
Real-time convolution #2
    Your data are correct
```

```
Real-time convolution #3
    Your data are correct
```

```
The long convolution took 0.042554 secs
```

## Code

```
disp('-----')
disp('                               Code')
disp('-----')
type convolv_rt
```

-----  
Code

---

```

-----
function y = convolv_rt(x, h)
    lh = length(h);
    hbuf = h(:)'; % make h a row vector
    x = [x(:); zeros(lh-1, 1)]; % pad x with zeros
    y = zeros(1, length(x)); % preallocate output array
    xbuf = zeros(lh, 1); % initialize input array as column vector
    for i = 1:length(x) % for each new value of x[n]
        % put x(i) into the buffer in reverse order
        xbuf(end - mod(i, lh)) = x(i);
        % set up the indexing into the hbuf and/or xbuf arrays here
        y(i) = hbuf * [xbuf(end - mod(i, lh):end); xbuf(1:end - mod(i, lh)
-1)]; % store output value
    end
end

```

*Published with MATLAB® R2023b*