Part 1: Problems from Gilat, Ch. 3.9

In your ~/`octave` directory, create a file `probs3.m` that contains `MATLAB/octave` code to solve the following problems.

As with the previous exercise, you should keep your editor open as you work through the problems, and ensure that you start `octave` from within your ~/`octave` directory. That way you can execute the commands in `probs3.m` (as you update and save the file) simply by typing `probs3` at the `octave` prompt.

```
octave:1> probs3
```

Note: If you would rather *not* have the output from `octave` piped through `more` as your script executes, put the following at the beginning of `probs3.m`

```
more off
```

1. For the function

$$y = \frac{(2x^2 - 5x + 4)^3}{x^2},$$

calculate the value of y for the following values of x: $-2, -1, 0, 1, 2, 3, 4, 5$ using element-by-element operations.

2. For the function

$$y = 5\sqrt{t} - \frac{(t+2)^2}{0.5(t+1)} + 8,$$

calculate the value of y for the following values of t: $0, 1, 2, 3, 4, 5, 6, 7, 8$ using element-by-element operations.

6. The position as a function of time $(x(t), y(t))$ of a projectile fired with a speed of v_0 at an angle θ is given by

$$\begin{aligned} x(t) &= v_0 \cos\theta \cdot t \\ y(t) &= v_0 \sin\theta \cdot t - \frac{1}{2} g t^2 \end{aligned}$$

where $g = 9.81\text{m/s}^2$ is the gravitation of the Earth. The distance r to the projectile at time at time t can be calculated by $r(t) = \sqrt{x(t)^2 + y(t)^2}$. Consider the case where $v_0 = 100\,\text{m/s}$ and $\theta = 79°$. Determine the distance r to the projectile for $t = 0, 2, 4, \ldots, 20$ s.

8. Define x and y as the vectors $x = 2, 4, 6, 8, 10$ and $y = 3, 6, 9, 12, 15$. Then use them in the following expression to calculate z using element-by-element calculations.

$$z = \left(\frac{y}{x}\right)^2 + (x+y)^{\left(\frac{y-x}{x}\right)}$$

10. Show that

$$\lim_{x \to 0} \frac{e^x - 1}{x} = 1$$

Do this by first creating a vector x that has the elements: $1, 0.5, 0.1, 0.01, 0.001, 0.00001$ and 0.0000001. Then create a new vector y in which each element is determined from the elements of x by $(e^x - 1)/x$. Compare the elements of y with the value 1 (use `format long` to display the numbers).

12. Use `octave` to show that the sum of the infinite series

$$\sum_{n=0}^{\infty} \frac{1}{(2n+1)(2n+2)}$$

converges to ln 2. Show this by computing the sum for

1. $n = 50$
2. $n = 500$
3. $n = 5000$

For each part, create a vector n in which the first element is 0, the increment is 1 and the last term is 50, 500 or 5000. Then use element-by-element calculation to create a vector in which the elements are

$$\frac{1}{(2n+1)(2n+2)}$$

Finally, use the function `sum` to add the terms in the series. Compare the values obtained in parts 1, 2 and 3 to ln 2.

18. Solve the following system of five linear equations:

$$\begin{aligned} 1.5x - 2y + z + 3u + 0.5w &= 7.5 \\ 3x + y - z + 4u - 3w &= 16 \\ 2x + 6y - 3z - u + 3w &= 78 \\ 5x + 2y + 4z - 2u + 6w &= 71 \\ -3x + 3y + 2z + 5u + 4w &= 54 \end{aligned}$$

Part 2: Basic 2D plotting with `octave`

Using the script file `~phys210/octave/plotex.m` as a guide, as well as information on the `plot` command available via `doc plot` (or through the on-line `octave` manual available from main Course Notes page), make a figure that shows plots of $\exp(-(x-2)^2)$, $\sin(2x)\cos^2(7x)$ and $\tanh(x - x^3)$ for $-6 \le x \le 6$.

Your figure should use 2000 uniformly distributed points on the plotting interval, and the three functions should be drawn with red, green and blue lines respectively. Include axes labels and a title of your own choosing. Finally, include a command to save a hardcopy of your figure as the (encapsulated) color Postscript file `myplot.ps`.

You should prepare the `octave` commands to make this figure in a script file `~/octave/myplot.m`.

Part 3: (Pseudo)-Random Numbers

Answer the following questions in an `octave` script file `~/octave/myrand.m`

1. Demonstrate that the mean value of the random numbers generated by `rand` approaches 0.5 as the length, n, of the random number sequence approaches ∞. Do this by computing the mean value for sequences of length $n = 10, 10^2, 10^3, 10^4, 10^5, 10^6$ and 10^7. Try to make your solution of the problem as concise as you can.

2a. Demonstrate that the mean value of the random numbers generated by `randn` approaches 0.0, and the standard deviation approaches 1.0, as the length, n, of the random number sequence approaches ∞. Do this by computing the mean value and standard deviation for sequences of length $n = 10, 10^2, 10^3, 10^4, 10^5, 10^6$ and 10^7. Again, try to make your solution of the problem as concise as possible. What do you observe about the results?

2b. Use `octave`'s `hist` function (type `doc hist` for usage information) to plot histograms with 1000 bars for the case of a million random numbers generated by `rand` and `randn` respectively.