Please provide complete and well-written solutions to the following exercises.

Due October 27, 1159PM PST, to be uploaded as a single PDF document to blackboard (under the Assignments tab).

Homework 8

Exercise 1. Let $f(x) := e^{2x}$ for all $x \in \mathbf{R}$. Let p_n be a polynomial of degree n that interpolates f on $[-1,1]$ at the $n+1$ roots of the Chebyshev polynomial T_{n+1} on $[-1,1]$. Find the smallest n such that

$$|f(x) - p_n(x)| < 10^{-6}, \qquad \forall\, x \in [-1,1]$$

Exercise 2. Let $h > 0$. Let $f\colon [-h,h] \to \mathbf{R}$ be continuous. Let p_2 be the (unique) polynomial of degree at most 2 such that

$$p_2(h) = f(h), \qquad p_2(-h) = f(-h), \qquad p_2(0) = f(0).$$

Show that

$$\int_{-h}^{h} p_2(t)dt = \frac{h}{3}\Big(f(h) + f(-h) + 4f(0)\Big).$$

Exercise 3 (Adaptive Quadrature)**.** The NCM package function `quadtx` is a simplified version of Matlab's built-in integration function `quad`. (To view the code of `quadtx` use the command `edit quadtx`. Similarly, `edit quad` should show you the source code for the function `quad`.) For example, the command `quadtx(@(x)(cos(x))^2,0,4*pi)` approximates $\int_0^{4\pi}(\cos(x))^2dx$. More generally, the program `quadtx` starts by evaluating the given function $f\colon [a,b] \to \mathbf{R}$ with two different Simpson's rule evaluations (one using three points, and another using five points, each equally spaced on the interval). (In the code, these two evaluations are denoted `Q1` and `Q2`.)

If these two different Simpson's rule evaluations are closer than 10^{-6} (the default value of `tol`), then the program believes it has succeeded in estimating $\int_a^b f(x)dx$. So, the program outputs a combination of these two Simpson's rule evaluations, which happens to be a sixth order Newton-Cotes formula (in the code this is `Q2 + (Q2 - Q1)/15`).

If these two different Simpson's rule evaluations are not closer than 10^{-6}, then `quadtx` repeats the above Simpson's rule procedure on a smaller subinterval, and then iterates. This is done via a recursive call to the function `quadtxstep`. Note the recursive nature of this program, since the function `quadtxstep` calls itself. Also, note that `varargin` is used often in `quadtx`. This command allows a variable number of arguments to be input to a function.

The recursive use of Simpson's rule can be visualized with the `quadgui` command, after clicking "auto." Function evaluations are depicted as blue dots, and the total number of function evaluations is displayed at the top of the plot.

- Run the programs `quadtx(@(x)x.^3,0,1)` and `quadgui(@(x)x.^3,0,1)`. How many function evaluations are used to estimate $\int_0^1 x^3 dx$? What is the absolute error of the estimation?
- Run the programs `quadgui(@(x)x.^5,0,1)` and `quadgui(@(x)x.^5,0,1,10^(-8))`. (Also use `quadtx` with the same arguments.) In each case, how many function evaluations are used to estimate $\int_0^1 x^5 dx$? What is the absolute error of the estimation?
- Run the program `quadtx(@(x)(cos(x)).^2,0,4*pi)`. How many function evaluations are used to estimate $\int_0^{4\pi} (\cos(x))^2 dx$? What is the absolute error of the estimation? Explain what happened. Does `quad(@(x)(cos(x)).^2,0,4*pi)` produce the same output? Explain why or why not.
- Describe a nonnegative function $f\colon [0,1] \to [0,1]$ such that the Matlab built-in command `quad` has the same error as in the previous part of this problem. That is, find f such that the command `quad(@(x) f(x),0,1)` outputs 1 and has absolute error at least 10^{-4}.

Exercise 4.

- Using the textbook program `quadtx`, try to integrate the function $\frac{1}{3x-1}$ from $x = 0$ to $x = 1$. Do you get an error? If so, explain why the error happened.
- Find a function $f\colon [0,1] \to [0,\infty)$ such that $\lim_{x\to 0^+} f(x) = \infty$ and with $\int_0^1 f(x)dx$ finite. Can the programs `quadtx` and `quad` evaluate your integral with good relative accuracy?
- Find an interval $[a,b]$ and a function $f\colon [a,b] \to \mathbf{R}$ that exceeds the maximum function evaluation count (i.e. produces the maximum function count warning) both for `quadtx` and `quad` when trying to estimate $\int_a^b f(x)dx$.

Exercise 5. Recall that we defined $U_0, U_1, \ldots$ to be the Chebyshev polynomials of the second kind, where $U_n(x) := \frac{\sin((n+1)\cos^{-1} x)}{\sin \cos^{-1} x}$ for any $x \in (-1,1)$.

- Show that $U_0, U_1, \ldots$ satisfy the recursion
$$U_{n+1}(x) = 2xU_n(x) - U_{n-1}(x), \qquad \forall\, n \geq 1, \quad \forall\, x \in (-1,1),$$
where $U_0(x) = 1$ and $U_1(x) = 2x$.
- Show that
$$\frac{d}{dx} T_n(x) = nU_{n-1}(x), \qquad \forall\, n \geq 1, \quad \forall\, x \in (-1,1),$$
where $T_0, T_1, \ldots$ are the Chebyshev polynomials (of the first kind).

Exercise 6. It is known that
$$\pi = \int_{-1}^{1} \frac{2}{1+x^2} dx.$$
(You can compute this integral using $(d/dx)\tan^{-1}(x) = 1/(1+x^2)$.)

- In Matlab, program your own trapezoid rule to estimate the integral $\int_{-1}^{1} \frac{2}{1+x^2} dx$. Plot the relative error of the integral estimate versus the number n of points used in the trapezoid rule. Do you see any evidence of numerical errors? (You might need to take n to be quite large, e.g. around $30,000$.)

- Use `quadtx` to estimate $\int_{-1}^{1} \frac{2}{1+x^2} dx$. Make a table recording the integral estimates for various tolerance values, including the estimated integral value `Q`, the function evaluate count `fcount`, and the relative error. (For example, consider tolerances of the form 10^{-k} where $k \in \{1, 2, 3, \ldots, 12\}$.) The first two rows of the table might look like this:

```
    tol               Q            fcount     Relative Error
********************************************************
  1e-001      3.14211764705882        9     1.671e-004
  1e-002      3.14211764705882        9     1.671e-004
```

  Does the relative error decrease as the tolerance decreases?

Exercise 7 (Optional)**.** Read some more about Shamir's Secret Sharing, as we discussed in class.