# CSC260S: Scientific, Graphical and Symbolic Computation
## Department of Computer Science
## University of Toronto

Instructor: Eugene Fiume | Assignment 2 | Due Date: 10 March 1995

You will be using Maple to answer questions 1 and 2, which will be worth about 20% and 30% of your grade, respectively. Question 3 is to be answered using OOT/C and is worth about 50%.

1) Review Appendix A of Chapter 2. Don't worry if you haven't had any linear algebra. Just follow the appendix and type **`?matrix`** and **`?linalg[inverse]`** in Maple to learn how to define matrices and take their inverses. Recall that our matrix form for polynomials is

$$p(t) \;=\; T\,M\,V,$$

where $T = [t^n \;\; t^{n-1} \;\; \cdots \;\; t^1 \;\; 1]$ is the monomial basis, $M$ is our change-of-basis, and $V$ is the desired set of ''observables''. Our polynomials are to be parameterised on [0,1]. To solve this question, write a Maple script to form a matrix $B$ as in Appendix A that accumulates the simultaneous constraints on $p(t)$. Then invert $B$ to get $M$. The two cases are:

(a) We have two points $P_0$ and $P_1$, two tangents $T_0$ and $T_1$, and two second derivatives $A_0$ and $A_1$. The resulting polynomial $p(t)$ must be such that

$$\begin{aligned} p(0) &= P_0, & p(1) &= P_1, \\ \dot{p}(0) &= T_0, & \dot{p}(1) &= T_1, \\ \ddot{p}(0) &= A_0, & \ddot{p}(1) &= A_1, \end{aligned}$$

where $\dot{p}$ is the first derivative of $p$ w.r.t. $t$ and $\ddot{p}$ is the second derivative.

b) We have four points $P_0$, $P_1$, $P_2$, $P_3$. The resulting polynomial $p(t)$ must be such that

$$\begin{aligned} p(0) &= P_0, & p(1) &= P_3, \\ \dot{p}(0) &= 3(P_1 - P_0), & \dot{p}(1) &= 3(P_3 - P_2). \end{aligned}$$

In your solutions, indicate the degree of the resulting polynomial, print the initial matrix $B$ of constraints, the basis matrix $M$, the basis functions $T\,M$, and plot the blending functions on [0,1]. Then describe how you would use each kind of curve in a piecewise setting, how values should be shared across curve segments, and what degree of continuity can be (easily) attained across curve segments. You are welcome to order the observables in $V$ as you like. In fact it is worthwhile to change the order around to see what effect this has on $M$.

2 You will be computing polynomials using forward differencing and studying its error behaviour in a specific context. A file of Maple procedures and stubs will be provided.

First, you will write Maple code to evaluate polynomials of arbitrary degree using forward differencing. Let the Maple sequence

$$\mathbf{A} = [a_n, a_{n-1}, \cdots, a_0]$$

(note the order of the $a_i$) denote a polynomial

$$p(t) \;=\; \sum_{i=0}^{n} a_i t^i.$$

Then **`A`** uniquely defines a polynomial in monomial form on which we can perform forward differencing. The provided code will ask you to implement several procedures, and you will ultimately write a procedure called **`evalpoly(A,t0,t1,k)`** that outputs a sequence of **`k`**+1 evenly-spaced points on the polynomial denoted by **`A`**, on the interval **`[t0,t1]`**. For example, if

```
A = [-1,3,4,-8,4,3,1];
```

then through the provided routine **`poly`**, this defines the polynomial

```
> poly(A,t);
```

$$- t^6 + 3 t^5 + 4 t^4 - 8 t^3 + 4 t^2 + 3 t + 1$$

and

```
> evalpoly(A,0,1,5);
```

gives

$$[[0,1], [1/5, \frac{26614}{15625}], [2/5, \frac{38391}{15625}], [3/5, \frac{50266}{15625}], [4/5, \frac{65989}{15625}], [1,6]] .$$

Normally **`k`** is larger than **`5`** but the expressions get too large to fit on the page. A procedure called **`plotpoly`** found in the same file will exercise your code by comparing points computed using forward differences and those computed more carefully. Once you have exercised your code with **`plotpoly`** for a variety of polynomials of different degrees (**`A`**), parametric intervals (**`t0`** and **`t1`**), and number of points (**`k`**), you can move on the the next part of this question.

For the second part, recall that the $n$ forward differences of a function $f$ are denoted

$$f, \quad \Delta f, \quad \Delta^2 f, \quad \Delta^3 f, \quad \cdots, \quad \Delta^n f.$$

If $f$ is a polynomial of degree $n$, then $\Delta^n f$ is a constant. Now, suppose there is a small error of $\varepsilon > 0$ made *only* in $\Delta^n f$, so that we replace the sequence of forward differences above by

$$f, \quad \Delta f, \quad \Delta^2 f, \quad \Delta^3 f, \quad \cdots, \quad (\Delta^n f + \varepsilon).$$

Make the (very small) change required to model this error in **`evalpoly`**. Investigate the effect of this perturbation for small values such as $\varepsilon = 10^{-8}$. If you plot the difference between accurately computed points on the curve and the perturbed points, where is the error maximised? On what does the size of the error depend? Is the domain **`[t0,t1]`** over which the computation takes place relevant to the error made? Given your investigation, and assuming small errors do occur in the initial computation of forward differences, would you recommend using forward differencing to compute arbitrary polynomials? If so, explain why. If not, outline a class of values for which **`evalpoly(A,t0,t1,k)`** gives good results.

3) You will be adding to an interactive curve generation package in OOT/C. As in the first assignment, you will be given a stub (Spline.t or Spline.c) that does a lot of the preparatory work for you. Your goal in this question will be to implement two polynomial curve evaluation techniques: Horner’s rule and computation of the monomial coefficients that we called $A$ above. This will give you a chance to compare these routines to direct curve evaluation, which will already be implemented for you. You must also implement the cubic Bezier and four-point cubic (Lagrange) interpolants. The basis matrix for the Bezier form comes from question 1b above, and the four-point interpolant is described on p94 (or 118) of the book. A posting will provide more details. A bonus will be awarded if you enhance the curve package to allow users to change the curve interactively (by allowing users to change the position of points they have already entered). Students interested in doing this should contact me.