# Homework 1 — Simple ML programs — assigned Friday 3 February — due Sunday 12 February

### Reading assignment

Read Chapters 1, 2, and 3 of *ML for the Working Programmer*.

## 1.1 Simple ML programs (5pts) [A.1]

Write a function `innerproduct: (real * real * real) * (real * real * real)- > real` to compute the inner product of two real vectors in $\mathbf{R}^3$ represented as ML tuples.

## 1.2 Simple ML programs (5pts) [A.1]

Write a function `innerproduct: real list * real list -> real` to compute the inner product of two real vectors in $\mathbf{R}^n$ represented as ML lists.

## 1.3 Pragmatics of integer arithmetic (15pts) [A.3]

The factorial function may be defined as:

```
fun fact n = if n <= 1 then 1 else n * fact (n-1)
```

which allows the binomial coefficient $\binom{n}{m}$ to be defined as:

```
fun binom (n, m) = (fact n) div (fact m * (fact (n-m)))
```

A shortcoming of this definition is that the computation may trigger integer overflow even when the final result fits in an integer. (We assume here that we are using SML/NJ, in which the type `int` has a finite range.)

1. (5pts) Experimentally determine the range of integers.
2. (10pts) Write a different definition of `binom` that avoids this problem.

## 1.4 Concrete mathematics (20pts) [A.3]

Let

$$f(p,k,i,j) = \sum_{s=0}^{p} (-1)^s \frac{\binom{p}{s}\binom{p-1}{j-1}\binom{k}{s+i-j}}{\binom{p+k-1}{s+i-1}}$$

where $p$,$k$,$i$, and $j$ are integers such that $2 \leq p$, $1 \leq k$, $1 \leq i \leq k$, and $1 \leq j \leq p$.

1. (10pts) Express the computation of $f$ as an ML function
   `f: {p:int, k:int, i:int, j:int} -> real`.

2. (10pts) Evaluate $f(p,k,i,j)$ for various values of $p$,$k$,$i$, and $j$, form a conjecture about the value of $f(p,k,i,j)$, and prove it.

### 1.5 Using lists for sets: writing recursive functions over lists (25pts) [A.1; K.2.2]

Let us use the ML type `int list` to represent sets of integers. The representation invariants are that there are no duplicates in the list, and that the order of the list elements is increasing.

1. (5pts) Write an ML function `union: int list * int list -> int list` that takes two sets and returns their union.
2. (5pts) Write an ML function `intersection: int list * int list -> int list` that takes two sets and returns their intersection.
3. (5pts) Write an ML function `difference: int list * int list -> int list` that takes two sets and returns their set difference.
4. (5pts) Write an ML function `equal: int list * int list -> bool` that takes two sets and returns `true` if and only if the two sets are equal.
5. (5pts) Write an ML function `powerset: int list -> int list list` that takes a set $S$ and returns its powerset $2^S$. (The powerset $2^S$ of a set $S$ (sometimes written $P(S)$) is the set of all subsets of $S$.) Note that the result uses the ML type `int list list` to represent sets of sets of integers. Here the representation invariant is that there are no duplicates in the list; the order of the sublists is immaterial.

### 1.6 Modelling (40pts) [A.1; K.2.2]

In this problem we consider a situation that arises in some recent research in microsystems, its mathematical model, and its representation as ML code.

#### 1.6.1 The device

Consider the microfluidic mixing network [JDC+00, DCJW01] shown schematically in Figure 1. The network consists of $k$ stages, and has $p$ inlets and $p+k$ outlets. Each stage splits $n$ flows into $n+1$ flows, for $n = p,\ldots,p+k-1$. It is assumed that the channels are fabricated with precision that allows all channel widths at the same level, and consequently all flows at the same level, to be assumed equal. The splitting of inlet flows in a stage is simple, owing to perfectly laminar flow. Each inlet flow is split into exactly two outlet flows. Each outlet flow is a combination of exactly two inlet flows, except for the two extremal outlets, each of which carries the unmixed flow from its corresponding extremal inlet. After the splitting, complete mixing occurs in the long and narrow serpentine channels.

#### 1.6.2 The model

The transfer matrix for a flow-splitting stage with $n$ inlets and $m$ outlets describes how the flows are split and mixed. If the concentrations of a particular solute in the $n$ inlets are grouped into a column vector $\mathbf{c}^{\text{in}}$

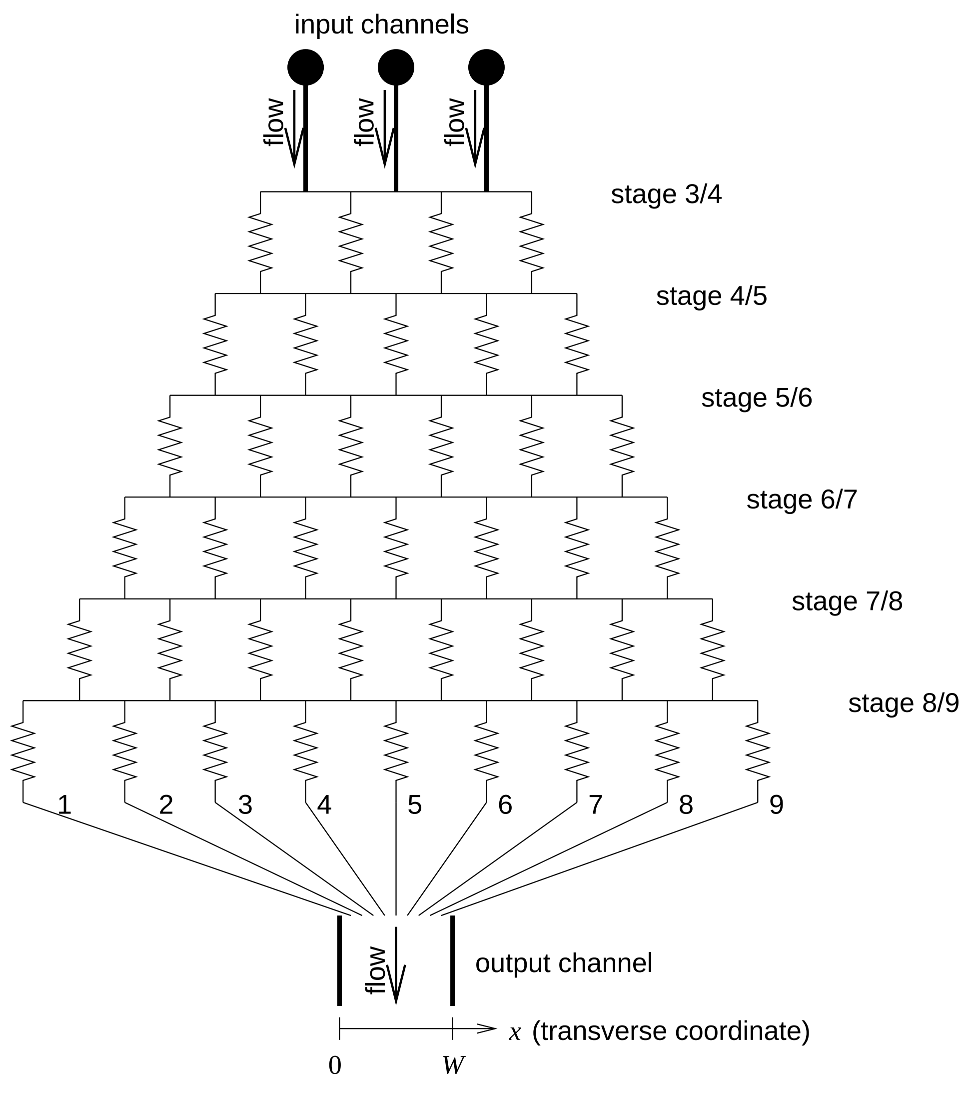


Figure 1: A microfluidic mixing network (after Jeon et al. [JDC$^+$00]) with $p = 3$ inlets to the first stage, $k = 6$ stages, and $p + k = 9$ outlets from the final stage.

of $n$ values, and similarly the concentrations in the $m$ outlets are grouped into a column vector $\mathbf{c}^{\text{out}}$ of $m$ values, then we have $\mathbf{c}^{\text{out}} = \mathbf{T}_{m,n}\mathbf{c}^{\text{in}}$, where $\mathbf{T}_{m,n}$ is the transfer matrix.

Each row of a transfer matrix gives the composition of a single outlet flow in terms of the inlet flows. Conversely, each column of a transfer matrix describes how a single inlet flow is distributed across the outlet flows.

From fluid flow considerations it is evident that the transfer matrix for a single stage with $p$ inlets and $p+1$ outlets is a $(p+1)\times p$ band matrix:

$$\mathbf{M}_{p+1,p} = \begin{bmatrix} 1 & 0 & \cdots & & & \\ \frac{1}{p} & \frac{p-1}{p} & 0 & \cdots & & \\ 0 & \frac{2}{p} & \frac{p-2}{p} & 0 & \cdots & \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ & \cdots & 0 & \frac{p-2}{p} & \frac{2}{p} & 0 \\ & & \cdots & 0 & \frac{p-1}{p} & \frac{1}{p} \\ & & & \cdots & 0 & 1 \end{bmatrix}$$

### 1.6.3 The code

We adopt the type `real list` as our representation of column vectors and the type `real list list` as our representation of matrices.[1]

In all that follows, assume that matrices and vectors are conformant.

1. (5pts) Write a function `mvp: real list list * real list -> real list` that computes a matrix-vector product.
2. (5pts) Write a function `mmp: real list list * real list list -> real list list` that computes a matrix-matrix product.
3. (10pts) Write a function `mixsucc: int -> real list list` such that `mixsucc p` equals $\mathbf{M}_{p+1,p}$.
4. (5pts) Compute `mvp (mixsucc p, v)` for some representative vectors $v$ and small $p$.
5. (10pts) The effect of successive stages is given by the product of matrices $\mathbf{M}_{p+1,p}$ for successive values of $p$. Let $\mathbf{T}_{p+k,p} = \mathbf{M}_{p+k,p+k-1}\cdots\mathbf{M}_{p+1,p}$. Write a function `mixcomp: int * int -> real list list` such that `mixcomp (p, k)` equals $\mathbf{T}_{p+k,p}$.
6. (5pts) Compute `mixcomp (3, 6)`.

## References

[DCJW01] Stephan K. W. Dertinger, Daniel T. Chiu, Noo Li Jeon, and George M. Whitesides. Generation of gradients having complex shapes using microfluidic networks. *Analytical Chemistry*, 73:1240–1246, 2001.

[1] Note that there are better representations available in the Standard ML Basis Library.

[JDC+00] Noo Li Jeon, Stephan K. W. Dertinger, Daniel T. Chiu, Insung S. Choi, Abraham D. Stroock, and George M. Whitesides. Generation of solution and surface gradients using microfluidic systems. *Langmuir*, 16:8311–8316, 2000.

## How to turn in

Turn in your code by running

*˜marron/cs451/handin your-file*

on a regular UNM CS machine. You should use whatever filename is appropriate in place of your-file. Whole directories may be submitted as well.

Include the following statement with your submission, signed and dated:
*I pledge my honor that in the preparation of this assignment I have complied with the University of New Mexico Board of Regents' Policy Manual.*