Problem 1: The Gregorian calendar puts 365 days in every ordinary year, 366 days in every leap year. Leap years are those years evenly divisible by 4 except for century years (the years divisible by 100), which are leap years only if divisible by 400 too. For example, 1600 and 2000 were leap years, but 1700, 1800 and 1900 were not, nor will 2100 be.

Assuming that the world retains the Gregorian calendar forever, prove that the probability that future Christmas Days (25 Dec.) will fall on Wednesdays is NOT exactly $1/7$.

Proof 1: Every 400 years the Gregorian calendar repeats its pattern of days-of-the-week *vs.* days-of-the-month. This happens because the number of days in 400 years, namely

$$400 \times 365 + (100 - 3) = 146097 = 7 \times 20871,$$

is evenly divisible by 7. (The term $100 - 3$ accounts for the leap years, one every four years except for the three century years not divisible by 400.) In every 400 years, Christmas Day falls on a Wednesday N times for some integer N. As eons go by, the relative frequency with which Christmas has fallen on a Wednesday must tend to $N/400$, its probability; but this fraction cannot simplify to $1/7$.

This problem comes from an old Putnam exam. Aside from dynastic and religious considerations, there are technical reasons to doubt that the Gregorian calendar, instituted in Catholic countries in 1582 by skipping from 4 to 15 Oct., and in the British Empire in 1752 by skipping from 2 to 14 Sept., will persist forever. The Gregorian year is too long, about a day too long in 4000 years. This could be remedied by denying leap-year status to millennial years divisible by 4000 as well as century years not divisible by 400. Were this remedy adopted, 6000 and 10000 would be leap-years (as was 2000), but not 3000, 4000, 5000, 7000, 8000 nor 9000. Such a remedy would keep the seasons in their now familiar places in the calendar for more than 16000 years. If this modified Gregorian calendar were adopted and kept forever, what would be the probability that Christmas will fall on a Wednesday? For MATLAB programs to predict days of the week see

<www.cs.berkeley.edu/~wkahan/daydate>

Problem 2: A standard deck of 52 playing cards will be shuffled to randomize them thoroughly. Then they will be dealt face up one at a time until the first ace appears. The next card— the one following the first ace— will be put in an envelope. Of two cards, the ace of spades and the two of clubs, which is more likely to go into the envelope?

The future tense has been used above because the probability of finding a card in the envelope is zero if it has already appeared face up. The probability we wish to compute pertains to our state of knowledge before any card has been dealt.

At first sight, the two of clubs seems the more likely to go into the envelope because the first ace may be the ace of spades. However, we must take account of the possibility that the two of clubs will precede the first ace. Since each of the four aces is as likely as any other to be first, the probability that the first ace will be the ace of spades is $1/4$. In the set of five cards consisting of the two of clubs and the four aces, the probability is $1/5$ that the first to appear will be the two of clubs. This seems to imply that the two of clubs is more likely than the ace of spaces to be put in the envelope. But this paragraph's reasoning is wrong. What is the right answer to the problem's question?

Solution 1: Shuffling the cards "to randomize them thoroughly" implies that every ordering of the 52 cards is as likely as any other ordering; there are $52!$ different orderings. In how many of these orderings does the ace of spades follow immediately after the first ace? We can count them by removing the ace of spades, ordering the 51 remaining cards, and then putting the ace

of spades back immediately after the first ace; there are $51!$ such orderings. Therefore the probability that the ace of spades will be found in the envelope is $(51!)/(52!) = 1/52$. The same argument yields the same probability $1/52$ for finding the two of clubs (or any other card) in the envelope. Surprisingly, each of the 52 cards in the deck is as likely as any other to go into the envelope.

Prof. M. Klass suggested this problem. Probabilistic arguments are easy to bungle. This is no excuse for avoiding them. Instead, practice them until you get them right. Aristotle, tutor to Alexander the Great, is said to have advised him …

" Consider what is probable, even if others deem it impossible, before considering possible improbabilities."

Don't confuse this advice with the better-known advice Aristotle offered playwrights and novelists:

" Rather choose (for dramatic purposes) a plausible impossibility than an improbable possibility." (*Poetics* §24)

Problem 3: A dart board is partitioned into disjoint regions each assigned an integer, and then three darts are thrown at the board at random independently, and the three integers so selected are added. Let p be the probability that their sum is divisible by 3. Find the minimum of p over all possible partitionings and assignments.

Solution 3: The minimum p is $1/4$. Here is why: Let x, y, z be the probabilities that the number recorded for a throw of a dart yield a remainder of $0, 1, 2$ respectively when divided by 3. The sum of three such remainders can be divisible by 3 only if all three remainders are the same or all three are different. Taking their order into account, different remainders can be selected in six ways, so the probability of divisibility by 3 is $p := x^3 + y^3 + z^3 + 6xyz$. This yields $p = 1/4$ when $x = y = 1/2$ and $z = 0$. Therefore, we seek a proof that p cannot be less than $1/4$ when $x \ge 0$, $y \ge 0$, $z \ge 0$ and $x+y+z = 1$. We remove this last sum constraint by replacing p above by a redefined function with the same range of values:

$$p(x, y, z) := (x^3 + y^3 + z^3 + 6xyz)/(x+y+z)^3 .$$

Thus redefined, p is homogeneous of degree 0; *i.e.*, $p(x, y, z) = p(tx, ty, tz)$ for all $t \ne 0$. Therefore the three variables x, y, z need merely be assumed nonnegative and not all zero. In other words, the point (x, y, z) can be assumed to lie in a region P consisting of the closure of the positive orthant shorn of the vertex $(0, 0, 0)$, instead of lying in a triangular section of P.

Now Lagrange multipliers are not needed to locate the extrema of p in P. Strictly inside P, the stationary values of p are taken where all its partial derivatives vanish; there we find

$$x^2 + 2yz = y^2 + 2zx = z^2 + 2xy = (x+y+z)^2 \cdot p .$$

Subtracting pairs of the first three expressions and factoring yields

$$(x–y)(x+y – 2z) = (y–z)(y+z – 2x) = (z–x)(z+x – 2y) = 0 ,$$

from which follows necessarily $x = y = z$ and $p = 1/3$. This stationary value turns out to be neither a maximum nor minimum of p in P. On the faces of P, where just one variable (say z) vanishes, the stationary values of $p = (x^3 + y^3)/(x+y)^3 = (x^2 – xy + y^2)/(x+y)^2$ are taken where its partial derivatives with respect to the other two variables vanish. There we find

$$2x – y = –x + 2y = 2(x+y)\, p ,$$

which implies $x = y$ and $p = 1/4$. This turns out to be its minimum. On the edges of P two variables vanish and $p = 1$, which is its maximum in P.

Problem 4: This problem explores elementary probabilistic ideas. On a computer, a *Random Number Generator* U() can be construed as a "function" that takes no argument but produces, each time it is invoked, a random number independent of those it produces at other invocations. These random numbers are considered to be *Uniformly Distributed* in an interval, say the interval between 0 and 1, just when the probability of finding one of those random numbers in any particular subinterval is proportional to its width; *i.e.*, just when

$$\text{Probability}\{ u \le U() \le v \} = \max\{ 0, \min\{1, v\} - \max\{0, u\} \}$$
$$= v - u \quad \text{provided} \quad 0 \le u \le v \le 1 .$$

These random numbers are deemed *Independent* when no information about the next value of U() can be inferred from all its previous values. (Of course, a computer program cannot produce truly random numbers; they are actually "Pseudo-Random" and, if the program is constructed well enough, are practically impossible to distinguish from truly random numbers.) Given such a generator U(), uniformly distributed between 0 and 1, your task is to construct out of it a generator of random *Barycentric Coordinates* $\{\lambda, \mu, \nu\}$ distributed uniformly and independently over a triangle, and then to construct another generator of random barycentric coordinates $\{\kappa, \lambda, \mu, \nu\}$ distributed uniformly and independently over a tetrahedron.

What are Barycentric Coordinates? $\{\lambda, \mu\}$ are the barycentric coordinates of the center of gravity of a pair of masses λ and μ situated at the ends of a line segment joining, say, $\mathbf{x}$ and $\mathbf{y}$. As λ and μ vary through nonnegative values, the center of gravity $(\lambda\mathbf{x} + \mu\mathbf{y})/(\lambda+\mu)$ moves along the line segment. Similarly $\{ \lambda, \mu, \nu \}$ are the barycentric coordinates of the center of gravity of three masses λ, μ and ν situated at a triangle's three vertices, say $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$ respectively. As λ, μ and ν vary through nonnegative values, the center of gravity $(\lambda\mathbf{x} + \mu\mathbf{y} + \nu\mathbf{z})/(\lambda+\mu+\nu)$ moves throughout the triangle. And so on for a tetrahedron. The masses are assumed to sum to a positive number, customarily 1, to avoid having all of them 0. A distribution of random coordinates is *Uniform* just when the probability of finding a random point in any subregion is proportional to its area (in the plane) or its volume (in 3-space).

Solution 4: Given any three noncollinear vectors $\mathbf{x}, \mathbf{y}, \mathbf{z}$ that locate the vertices of a triangle T, the expression $\mathbf{t} := \lambda\mathbf{x} + \mu\mathbf{y} + \nu\mathbf{z}$ runs over the triangle's interior once as $\{ \lambda, \mu, \nu \}$ runs over all triples of positive numbers that add up to 1. In fact, given any point $\mathbf{t}$ inside T, the equation $\mathbf{t}-\mathbf{x} = (\lambda-1)\mathbf{x} + \mu\mathbf{y} + \nu\mathbf{z} = \mu(\mathbf{y}-\mathbf{x}) + \nu(\mathbf{z}-\mathbf{x})$ determines μ and ν uniquely, because $\mathbf{y}-\mathbf{x}$ and $\mathbf{z}-\mathbf{x}$ are linearly independent, and thus map *Bijectively* (one-to-one) the triangle

$$\Delta: \quad \mu \ge 0 \quad \text{and} \quad \nu \ge 0 \quad \text{and} \quad \mu + \nu \le 1$$

in the (μ, ν)-plane to triangle T. This bijective map is also *linear*, so if it maps a subregion Ω in Δ to a subregion R in T then (Area of Ω)/(Area of Δ) = (Area of R)/(Area of T).

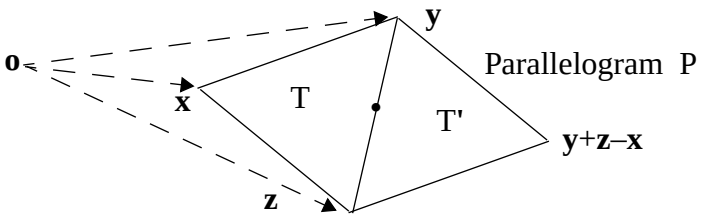
Now, *Random* barycentric coordinates $\{ \lambda, \mu, \nu \}$ are said to be distributed *Uniformly* over a triangle when, for every subregion R contained in the foregoing triangle T, the random point $\lambda\mathbf{x} + \mu\mathbf{y} + \nu\mathbf{z}$ falls into R with probability (Area of R)/(Area of T). Equivalently, since the random barycentric coordinates always satisfy $\lambda + \mu + \nu = 1$, they are distributed uniformly when the point (μ, ν) in the (μ, ν)-plane falls into a subregion Ω in Δ with probability (Area of Ω)/(Area of Δ). Our task is to invent formulas for λ, μ and ν that employ a given uniform and independent random number generator U().

Before devising three ways to use the given uniform random number generator U() to generate points uniformly over a triangle, let us solve a simpler problem:

How can we generate Random Barycentric Coordinates
{μ, ν} distributed Uniformly along a line segment?

Here μ and ν must be nonnegative and sum to 1. By setting ν := U() and μ := 1 – ν we obtain random barycentric coordinates that distribute random points $\mu\mathbf{x} + \nu\mathbf{y} = \mathbf{x} + \nu(\mathbf{y}–\mathbf{x})$ uniformly along the line segment joining points (vectors) **x** and **y**.

The foregoing construction inspired Ms Svetlana Zuev, a student taking Math. H90 in 1995, to observe that setting first u_1 := U() and then u_2 := U() produces a pair $\{u_1, u_2\}$ of independent random variables distributed uniformly over a square that can be mapped to a parallelogram P thus: For any noncollinear vectors **x**, **y**, **z** the point $\mathbf{x} + u_1\cdot(\mathbf{y}–\mathbf{x}) + u_2\cdot(\mathbf{z}–\mathbf{x})$ runs randomly and uniformly over the parallelogram P whose vertices are **x**, **y**, **z** and $\mathbf{y}+\mathbf{z}–\mathbf{x} = \mathbf{x} + (\mathbf{y}–\mathbf{x}) + (\mathbf{z}–\mathbf{x})$:



This P consists of two triangles: One of them is T with vertices **x**, **y**, **z**; the other is T' with vertices **y**+**z**–**x**, **z**, **y**. These congruent triangles share a common edge, the diagonal of P from **y** to **z**; and each triangle is the reflection of the other through the midpoint (**y**+**z**)/2 of P. Point **t'** in T' reflects through that midpoint to **t** = **y**+**z** – **t'** in T. If randomly generated points that fall into T' are relocated to their reflections in T, the random points distributed uniformly in P will then be mapped to random points distributed uniformly in T, as desired. The relocation is accomplished as follows, which is Svetlana Zuev's elegant solution of the given problem:

μ := U() ; ν := U() ; … each random, independent, uniform on [0, 1] .
if 1–μ < ν then do { μ := 1–μ ; ν := 1–ν } ; … { reflection from T' to T } if necessary.
λ := (1–μ) – ν . … and λ ≥ 0 .
… Now random λ, μ, ν are nonnegative and sum to 1, and
… λ**x** + μ**y** + ν**z** is distributed uniformly over the triangle T .

It's a neat process to execute and to explain, but can't be generalized so neatly to tetrahedra.

Rejection is an alternative to relocation, and gives rise to the following alternative solution for the given problem; simply discard any random point that falls into T' and try again to generate a random point in T :

Repeat { μ := U() ; ν := U() ; λ := 1 – μ – ν } until λ ≥ 0 .
… Now random λ, μ, ν are nonnegative and sum to 1, and
… λ**x** + μ**y** + ν**z** is distributed uniformly over the triangle T .

When the repeated block { μ := … – ν } generates a point that falls into T' it is rejected, which happens about as often as not on average. How many times will this rejection procedure invoke U() ? On average, we can expect U() to be invoked twice half the time, four times a quarter of

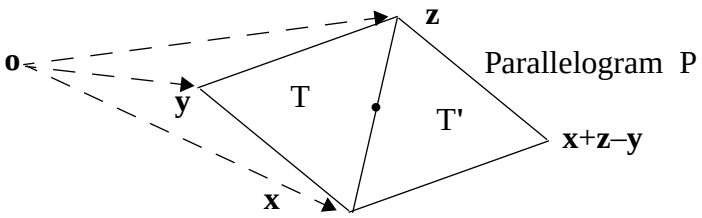
the time, six times an eighth of the time, …, amounting to a total of four invocations of U() on average. This means that the rejection procedure can be expected to take about twice as long on average as Svetlana Zuev's reflection procedure takes; worse, the rejection procedure will take a very long time on extremely rare occasions.

An inept attempt to speed up the rejection procedure might modify it thus:

$\mu := U()$; repeat { $\nu := U()$; $\lambda := 1 - \mu - \nu$ } until $\lambda \geq 0$.

This modification destroys the Uniformity of the distribution of random points in T ; can you see why? If not, run the modified procedure on a computer and plot the points it generates until the distribution's nonuniformity becomes evident.

Here is a third neat algorithm, and it does generalize to tetrahedra. Divide a parallelogram P whose vertices are x, y, z and x+z–y in cyclic order (different from the previous P) into two triangles along the diagonal from x to z . Now triangle T , with vertices x, y and z , is a reflection of the other triangle T' with vertices z, x+z–y and x , but in a less obvious way.



The point $\mathbf{t} := \mathbf{x} + \xi(\mathbf{y}–\mathbf{x}) + \eta(\mathbf{z}–\mathbf{y})$ runs over all of P as coordinates ξ and η both run independently between 0 and 1 . If $\eta \leq \xi$ then $\mathbf{t} = (1–\xi)\mathbf{x} + (\xi–\eta)\mathbf{y} + \eta\mathbf{z}$ is a positively weighted average of T's vertices and therefore lies in T ; if $\eta \geq \xi$ then $\mathbf{t}$ lies in T' . If ξ and η are random variables distributed independently and uniformly between 0 and 1 then $\mathbf{t}$ is randomly distributed uniformly over P , and so also over each of T and T' when not in the other. To map T' to T we simply swap ξ and η whenever $\eta \geq \xi$. Here is the algorithm:

$\xi := U()$; $\eta := U()$; ... two independent random variables uniform on [0, 1] .
if $\xi < \eta$ then swap $\{\xi, \eta\}$; … so $\xi \geq \eta$ now.
$\lambda := 1 - \xi$; $\mu := \xi - \eta$; $\nu := \eta$; ... all nonnegative.
... Now $\{ \lambda, \mu, \nu \}$ are barycentric coordinates (nonnegative and sum to 1) randomly
... and uniformly distributed over a triangle.

. . . . . .

As neatly as the foregoing three solutions for the given problem exploit its geometry, they are not the only neat solutions. Another scheme besides reflection and rejection deserves consideration, and it will generalize to a tetrahedron.

Given a random number generator U() whose Probability Density is *Uniform* on [0, 1] ,
i.e., if $0 \leq u \leq 1$ then Probability{ $u \leq U() < u+du$ } = $1 \cdot du$, otherwise it is 0 ,
let us derive from U() a random number generator V() whose Probability Density is *Linear* :
i.e., if $0 \leq v < 1$ then Probability{ $v \leq V() < v+dv$ } = $2v \cdot dv$; otherwise it is 0 .
One way to accomplish this is to set $V() := \sqrt{U()}$, as we'll see; and then we'll apply it to T .

More generally, consider a random variable X() whose *Probability Density* is f(x) , which means that

Probability{ $x \leq X() < x + dx$ } = $f(x) \cdot dx$ for infinitesimal dx .

Here f(x) must be nonnegative and integrable and its integral over the range of X() must be 1 .

Next define $F(x)$ to be the *Cumulative Distribution* function for $X()$, which means

$$\text{Probability}\{ -\infty < X() \le x \} = F(x) = \int_{-\infty}^{x} f(\xi) \cdot d\xi .$$

This $F(x)$ is a nondecreasing function rising from $F(-\infty) = 0$ to $F(+\infty) = 1$; and $F'(x) = f(x)$ wherever the derivative $F'(x)$ exists. For example, for the random variable $U()$ distributed uniformly between 0 and 1 we find the probability density is

$$f(u) = 1 \text{ if } 0 \le u \le 1, \text{ otherwise } f(u) = 0;$$

and the cumulative distribution of $U()$ is

$$\begin{aligned} F(u) &= 0 \text{ if } u \le 0, \\ &= u \text{ if } 0 \le u \le 1, \\ &= 1 \text{ if } u > 1. \end{aligned}$$

We shall need another function $M(x)$ which, for simplicity's sake, is presumed to be a strictly increasing continuous function that runs from $-\infty$ to $+\infty$ as x runs from $-\infty$ to $+\infty$. Then the equation $y = M(x)$ can be solved for $x = W(y)$; here $W(y)$ is also a strictly increasing continuous function whose domain and range are the whole real axis. The functions M and W are *Inverse* to each other in the sense that $W(M(x)) = x$ and $M(W(y)) = y$ for all real x and y.

Now consider a new random variable $Y() := M(X())$ obtained from a given random variable $X()$ with known density $f(x)$ and cumulative distribution $F(x)$; how is $Y()$ distributed? In other words, what density $g(y)$ and cumulative distribution $G(y)$ belong to $Y()$? Here they are:

$$\begin{aligned} G(y) &:= \text{Probability}\{ -\infty < Y() \le y \} \\ &= \text{Probability}\{ -\infty < M(X()) \le y \} \\ &= \text{Probability}\{ -\infty < X() \le W(y) \} = F(W(y)); \text{ and therefore} \\ g(y) &:= G'(y) = F'(W(y)) \cdot W'(y) = f(W(y)) \cdot W'(y) \quad \text{wherever all derivatives exist.} \end{aligned}$$

For example take $V() = \sqrt{U()}$ for $U()$ distributed uniformly on the interval $[0, 1]$, so its density $f(u) = 1$ for $0 \le u \le 1$. Here $M(u) := \sqrt{u}$ is specified on that interval. What M does elsewhere does not matter much; we can extend M to the whole real axis by setting $M(u) := \sqrt{|u|} \cdot \text{sign}(u)$. Its inverse is then $W(v) = v \cdot |v|$, so $W(v) = v^2$ on the interval $[0, 1]$, which is what matters. Now the density of $V()$ is $g(v) = f(W(v)) \cdot W'(v) = 2v$ on that interval, as was predicted above, and the cumulative distribution of $V()$ is $G(v) = F(W(v)) = v^2$. This random number generator $V()$ with its linear density will be used later to generate random points distributed uniformly over the triangle T.

There is another altogether different way to produce a random variable $V()$ with the same distribution. It requires two invocations of $U()$, instead of one plus a square root. We could write cryptically $V() := 1 - | U() + U() - 1 |$, but here is a more conventional rendering of the algorithm:

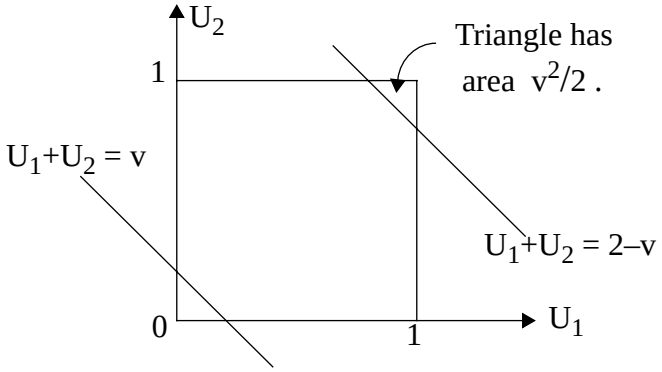
$$U_1 := U(); \quad U_2 := U(); \quad V := 1 - |U_1 + U_2 - 1| .$$

These U_1 and U_2 are independent random variables each distributed uniformly over the interval $0 \le U_j \le 1$, thereby distributing random points (U_1, U_2) uniformly over the square wherein $0 \le U_1 \le 1$ and $0 \le U_2 \le 1$. Here "uniformly" means that (U_1, U_2) falls into any region R contained in the square with probability equal to the area of R. Then how is $V()$ distributed? $0 \le V() \le 1$ since $0 \le U_1 + U_2 \le 2$. Moreover, for $0 \le v \le 1$,

$G(v) = \text{Probability}\{\ 0 \le V() \le v\ \} = \text{Probability}\{\ 1-v \le |U_1 + U_2 - 1| \le 1\ \}$

$= \text{Probability}\{\ 0 \le U_1 + U_2 \le v \quad \text{or} \quad 2-v \le U_1 + U_2 \le 2\ \} = v^2/2 + v^2/2 = v^2$.

The two contributions $v^2/2$ come from the triangles cut at two corners of the square by the previous inequalities.



Therefore random variables $V() := 1 - |U() + U() - 1|$ and $V() := \sqrt{U()}$ have the same linear distribution; to compute $V()$ on your computer choose whichever of these formulas runs faster.

. . . . . .

Let us resume the construction of random barycentric coordinates uniformly distributed over the triangle T. Suppose noncollinear vectors $\mathbf{x}$, $\mathbf{y}$, $\mathbf{z}$ locate the vertices of T, and consider the random vector $\mathbf{t}() := \mathbf{x} + V()\cdot\big((\mathbf{y}-\mathbf{x}) + U()\cdot(\mathbf{z}-\mathbf{y}) \big)$ wherein $U()$ is distributed uniformly and, independently, $V()$ is distributed with a linear density $(2V())$, both over the interval from 0 to 1. How is $\mathbf{t}()$ distributed?

$\mathbf{t}()$ sweeps out T as $U()$ and $V()$ sweep independently from 0 to 1 because the scalar coefficients of vectors $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$ sweep through all nonnegative barycentric coordinates with sum 1. Determining how densely $\mathbf{t}()$ is distributed over T raises a technical question that will be answered twice to illustrate first a geometrical technique and secondly an analytical technique.

First cut T into strips all of equal infinitesimal width and parallel to the edge of T opposite $\mathbf{x}$. Fixing v between 0 and 1 selects a strip, and then vector $\mathbf{x} + v\cdot(\mathbf{y}-\mathbf{x} + u\cdot(\mathbf{z}-\mathbf{y}))$ runs along this strip as u runs from 0 to 1. The random vector $\mathbf{x} + v\cdot(\mathbf{y}-\mathbf{x} + U()\cdot(\mathbf{z}-\mathbf{y}))$ is distributed uniformly along this strip, whose length and therefore area are proportional to v. Consequently $\mathbf{t}() = \mathbf{x} + V()\cdot\big(\mathbf{y}-\mathbf{x} + U()\cdot(\mathbf{z}-\mathbf{y})\big)$ is distributed uniformly along each strip at a density which, when accumulated over the whole strip, contributes to its probability an amount proportional to the density $2V()$ of $V()$, and thus proportional to the strip's area. Therefore the density of $\mathbf{t}()$ per unit area is the same in every strip, and thus uniform over T.

An analytic technique confirms that conclusion: As $U()$ and $V()$ sweep independently through an infinitesimal rectangle wherein $0 \le u \le U() < u+du \le 1$ and $0 \le v \le V() < v+dv \le 1$, which they do with probability $(1\cdot du)(2v\cdot dv)$, vector $\mathbf{t}()$ runs through an infinitesimal parallelogram with a vertex at $\mathbf{x} + v\cdot\big(\mathbf{y}-\mathbf{x} + u\cdot(\mathbf{z}-\mathbf{y})\big)$ and adjacent edges $v\cdot(\mathbf{z}-\mathbf{y})\cdot du$ and $\big(\mathbf{y}-\mathbf{x} + u\cdot(\mathbf{z}-\mathbf{y})\big)\cdot dv$. The area of this parallelogram is proportional to the determinant of the edges' coefficients of the independent constant vectors $\mathbf{z}-\mathbf{y}$ and $\mathbf{y}-\mathbf{x}$, namely $\det\left(\begin{bmatrix} v\cdot du & 0 \\ u\cdot dv & dv \end{bmatrix}\right) = v\cdot du\cdot dv$. Since this

infinitesimal parallelogram's area turns out proportional to the foregoing probability $2v\cdot du\cdot dv$ that **t**() will fall into it, **t**() must be distributed uniformly over the triangle T, as desired.

The desired barycentric coordinates $\{\lambda, \mu, \nu\}$ distributed uniformly over a triangle are the random coefficients of **x**, **y** and **z** in $\mathbf{t}() = \mathbf{x} + \big(\mathbf{y}–\mathbf{x} + (\mathbf{z}–\mathbf{y})\cdot U()\big)\cdot V()$, and are computed by this algorithm:

$U := U()$; $V := V()$; … independent uniform and linear random variables.
$\lambda := 1–V$; $\nu := V\cdot U$; $\mu := V–\nu$. … Don't confuse "*V*" with Greek letter "ν" (nu).

This is the last of the algorithms, namely
reflection, rejection, and a square root or sum to generate a linear distribution, devised to solve the given problem of barycentric coordinates distributed uniformly in a triangle.

. . . . . .

The foregoing problem is one of a sequence of such problems of which the next is to generate barycentric coordinates $\{\kappa, \lambda, \mu, \nu\}$ randomly, independently and uniformly distributed over the tetrahedron wherein they are nonnegative and add up to 1. The tetrahedron is swept out by $\kappa\cdot\mathbf{w} + \lambda\cdot\mathbf{x} + \mu\cdot\mathbf{y} + \nu\cdot\mathbf{z}$ provided the tetrahedron's vertices reached by vectors **w**, **x**, **y**, **z** are not coplanar; the differences **x**–**w**, **y**–**x**, **z**–**y** must be linearly independent vectors. A simple algorithm to compute those barycentric coordinates resembles the last technique described above; here it is:

First use one of the algorithms above to generate random barycentric coordinates
$\{\lambda', \mu', \nu'\}$ distributed uniformly and independently over the triangle;
$q := \sqrt[3]{U()}$; ... the independent random cube root has density $3q^2$ on $0 \le q \le 1$.
$\kappa := 1–q$; $\lambda := q\cdot\lambda'$; $\mu := q\cdot\mu'$; $\nu := q\cdot\nu'$.

A vindication of this algorithm is left to the reader.

Another algorithm to generate random barycentric coordinates $\{\kappa, \lambda, \mu, \nu\}$ uniformly distributed over the tetrahedron is analogous to the second neat algorithm above for the triangle, based upon a uniform distribution of $\mathbf{t} := \mathbf{w} + \xi\cdot(\mathbf{x}–\mathbf{w}) + \eta\cdot(\mathbf{y}–\mathbf{x}) + \zeta\cdot(\mathbf{z}–\mathbf{y})$ over a parallelepiped with a vertex at **w** and three adjacent edges **x**–**w**, **y**–**x** and **z**–**x**. The triple $\{\xi, \eta, \zeta\}$ must be randomly distributed uniformly and independently over the cube $0 \le \xi \le 1$, $0 \le \eta \le 1$ and $0 \le \zeta \le 1$. in (ξ, η, ζ)-space. This cube breaks into six tetrahedra of which one consists of those points (ξ, η, ζ) so ordered that $0 \le \xi \le \eta \le \zeta \le 1$; then the other five tetrahedra are obtained by permuting coordinates. Sorting $\{\xi, \eta, \zeta\}$ maps those five tetrahedra upon the first in a way that preserves the distribution's uniformity as follows:

$\xi := U()$; $\eta := U()$; $\zeta := U()$; … now (ξ, η, ζ) is uniform over the cube.
Sort (ξ, η, ζ) so that $\xi \le \eta \le \zeta$; … so (ξ, η, ζ) is uniform over the tetrahedron.
$\kappa := 1–\zeta$; $\lambda := \zeta–\eta$; $\mu := \eta–\xi$; $\nu := \xi$.

The verification that random barycentric coordinates $\{\kappa, \lambda, \mu, \nu\}$ are now distributed uniformly over a tetrahedron is left to the reader.

. . . . . .

The last two algorithms can be generalized further to distribute random barycentric coordinates uniformly over a simplex of arbitrarily high finite dimension. The *MATLAB* program `barand` below generates N random barycentric coordinate-sets distributed uniformly over a simplex of dimension K–1 . How can its correctness be proved? How should it be tested?

```
function  R = barand(K, N)
%   R = barand(K, N)  is a  KxN  array of random  Barycentric Coordinates
%   ( R >= 0  and  sum(R) == ones(1,N) )   distributed uniformly over the
%   (K-1)-Simplex,  which is a line segment when  K = 2 ,  a triangle when
%   K = 3 ,  a tetrahedron when  K = 4 ,  etc.  In general,  the vertices
%   of a simplex are the columns of  X = [x1 x2 x3 ... xK] ,  provided all
%   K-1  differences  x2-x1, x3-x2, x4-x3, ...  are linearly independent;
%   and then  X*barand(K, N)  scatters  N  points at random distributed
%   uniformly throughout the simplex.  For example,  to test  barand(3,N) ,
%                plot( ([0 1 i]*barand(3,N)).' , '.' )
%   should fill a triangle in the complex plane uniformly with dots if  N
%   is big enough.  To test  barand(4,N) ,  running
%                Z = [ 1 0 0 1 ;  0 1 -1 0 ]*barand(4,N) ;
%                plot( Z(1,:)' + (Z(1,:).*Z(2,:))'*i , '.' )
%   should scatter dots uniformly over a lens.

%   Program  barand  uses  Matlab's  rand(m,n)  to create an  m-by-n  array
%   of pseudo-random numbers distributed independently and uniformly between
%   0  and  1 .                                        W. Kahan,  9 Dec. 1996

K = round(K) ;  N = round(N) ;  % ...  Force  K  and  N  to be integers.
R = ones(K, N) ;  %...  Preallocate memory for  R .
if  K < 2 ,  return,  end
if  K == 2
    R(1,:) = rand(1,N) ;  % ...  uniformly distributed between  0  and  1 .
    R(2,:) = diff(R) ;
    return
  end
%  Svetlana Zuev's  method for  K = 3  distributes uniformly over a triangle:
k = [1 2] ;
R(k,:) = rand(2,N) ;  j = ( sum(R(k,:)) > 1 ) ;
R(k,j) = R([3; 3],j) - R(k,j) ;  % ...  = 1 - R(k,j)
R(3,:) = R(3,:) - sum(R(k,:)) ;  % ...  = 1 - sum(R(k,:))
%  If  K > 3 ,  distribute points over simplices of increasing dimensions:
for  j = 3:K-1
    k = [1:j] ;
    v = rand(1,N).^(1/j) ;
    R(k,:) = R(k,:).*v(ones(j,1),:) ;
    R(j+1,:) = R(j+1,:) - v ;  % ... = 1 - v = 1 = sum(R(k,:)) .
  end %...  of  barand

  ..................................
```

An extensive discussion of pseudo-random number generators, with many citations of the relevant literature, appears in vol. II of *The Art of Computer Programming* by D.E. Knuth, now in its third edition.

Problem 5: You dare not offend nor disappoint rich Great Uncle Willy lest he disown you. He has placed $2000 in your hands to bet on the outcome of the World Series, which is a baseball contest won by whichever of two teams first wins 4 games. Draws are not tolerated. Willy has chosen a team and given you $2000 before the Series starts. At the end of the Series he expects you to return to him either $4000 if the team he chose wins, or nothing if it loses.

But you can find nobody who will accept bets on the entire World Series. However, there are wagerers who will take on even-odds bets, in any amount(s), on each game individually. What strategy for placing bets on individual games will achieve the cumulative result Willy expects?

Solution 5: The betting strategy is laid out in the following array. It is organized into eight columns, one for each time just before the first, or just after the last, or between two games, of which at most seven will be played. These are the times at which money changes hands both to place a bet and to receive its winnings. At each time but the last, you must decide how much to bet and how much to hold; these amounts are entered into the array thus: `[Amount Bet]` / `[Amount Held]`

Whenever Willy's team wins, move to the array's next entry upward to the right; whenever his team loses, move to the array's next entry downward to the right. The last move must go either to an entry `[4000]` if Willy's team has just won the Series, or to an entry `[ 0 ]` if his team has just lost the Series. The array can be filled in from right to left because each entry's `[Amount Bet]` + `[Amount Held]` must match the `[Amount Held]` from a previous lost game as well as the sum `[Amount Held]` + 2`[Amount Bet]` from a previous win.

```
   Game #   1      2      3      4      5      6      7      #Won – #Lost
            :      :      :      :      :      :      :
            :      :      :      [4000]:      :      :       ...  4
            :      :      [ 250]/      :      :      :
            :      :      [3500]\      [4000]:      :       ...  3
            :      [ 500]/      [ 500]/      :      :
            :      [2750]\      [3000]\      [4000]:       ...  2
             [ 625]/      [ 750]/      [1000]/      :
 Start       [2000]\      [2000]\      [2000]\      [4000] ...  1
     [ 625]/      [ 750]/      [1000]/      [2000]/
     [1375]\      [1250]\      [1000]\      [  0 ]\       ...  0
 here        [ 625]/      [ 750]/      [1000]/      [    ]
             [ 750]\      [ 500]\      [  0 ]\      [  0 ] ... -1
                    [ 500]/      [ 500]/      [    ]
       Win          [ 250]\      [  0 ]\      [  0 ]       ... -2
 [ Bet]/                    [ 250]/      [    ]
 [Held]\                    [  0 ]\      [  0 ]              ... -3
       Lose                       [    ]
                                  [  0 ]                    ... -4
```

(Prof. Elwyn Berlekamp supplied this problem in 1995 and said this task had been set before job-seekers at a Wall Street brokerage in 1994.)

Problem 6: Show how to distribute nine points around the surface of a sphere in such a way that each point is equidistant from its four nearest neighbors.

Solution 6: To describe it we shall need two coordinate systems for Euclidean 3-space. First, the rectangular (x, y, z) coordinate system identifies a point p with a row-vector $\mathbf{p} = [x, y, z]$ of its coordinates. Second, the spherical coordinate system uses radial distance $r \ge 0$, azimuth angle θ, and elevation angle ϕ to locate a point. The coordinate systems are related:

$$x = r\cdot\cos(\theta)\cdot\cos(\phi)\ ,\quad y = r\cdot\sin(\theta)\cdot\cos(\phi) \quad\text{and}\quad z = r\cdot\sin(\phi)\ ;\ \text{and conversely}$$

$$r = \sqrt{(x^2 + y^2 + z^2)}\ ,\quad -\pi/2 \le \phi = \arcsin(z/r) \le \pi/2\ ,\quad -\pi \le \theta = 2\cdot\arctan(y/(x + \sqrt{(x^2+y^2)})) \le \pi\ .$$

(If $x < 0$, use $\theta = 2\cdot\arctan((\sqrt{(x^2+y^2)} - x)/y)$ to get better numerical accuracy.)

Since all the points in this solution lie on a sphere, say the unit sphere with $r = 1$, the angles θ and ϕ suffice to determine the vector $\mathbf{s}(\theta, \phi) := [\cos(\theta)\cdot\cos(\phi), \sin(\theta)\cdot\cos(\phi), \sin(\phi)]$ to the point $s(\theta, \phi)$ on the sphere. These angles could be used to describe our nine points on the sphere; but there would be 18 angles, and the 36 distances between points would have to be computed from these angles as was done on pp. 66-7 of *The College Math. Journal* **27** #1 (Jan. 1990), the source of this problem. Let's find a way to avoid most of that trigonometric computation.

Recall the scalar product $\mathbf{p}_1\cdot\mathbf{p}_2' = x_1\cdot x_2 + y_1\cdot y_2 + z_1\cdot z_2$ of two row-vectors $\mathbf{p}_j = [x_j, y_j, z_j]$. It figures in a vector's length $\|\mathbf{p}\| = \sqrt{(\mathbf{p}\cdot\mathbf{p}')}$ and in the distance $\|\mathbf{p}_1-\mathbf{p}_2\|$ between two points $\mathbf{p}_1$ and $\mathbf{p}_2$; in fact $\|\mathbf{p}_1-\mathbf{p}_2\|^2 = (\mathbf{p}_1-\mathbf{p}_2)\cdot(\mathbf{p}_1-\mathbf{p}_2)' = \|\mathbf{p}_1\|^2 + \|\mathbf{p}_2\|^2 - 2\cdot\mathbf{p}_1\cdot\mathbf{p}_2'$. Consequently distances between points $\mathbf{s}_1$ and $\mathbf{s}_2$ on the sphere depend only on their vectors' scalar products:

$$\|\mathbf{s}_1 - \mathbf{s}_2\|^2 = 2(1 - \mathbf{s}_1\cdot\mathbf{s}_2')\ .$$

The bigger the scalar product, the smaller the distance. Scalar products are so easy to compute in rectangular coordinates that these shall be used exclusively to solve our problem.

Angles involve rotations, which are represented by matrices in a rectangular coordinate system. To describe our solution we need one rotation and two reflections:

Rotation R	takes θ to $\theta + 2\pi/3$.	Consequently $R^{-1} = R^2$.
Reflection V	takes ϕ to $-\phi$.	Consequently $V^{-1} = V$.
Reflection H	takes θ to $-\theta$.	Consequently $H^{-1} = H$.

$$R = \frac{1}{2}\cdot\begin{bmatrix} -1 & \sqrt{3} & 0 \\ -\sqrt{3} & -1 & 0 \\ 0 & 0 & 2 \end{bmatrix}\ ,\quad V = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}\ ,\quad H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\ .$$

Evidently $RV = VR$, $HV = VH$, and $HRH = R' = R^{-1}$.

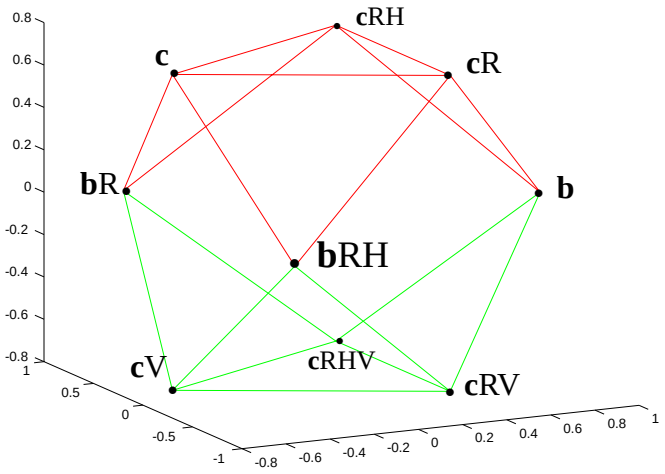
Nine points distributed as the problem requires are reached by vectors

$\mathbf{b} := [1, 0, 0] = \mathbf{b}V = \mathbf{b}H$,	$\mathbf{b}R$,	$\mathbf{b}R^{-1} = \mathbf{b}RH$,
$\mathbf{c} := [-2/3, 0, \sqrt{5}/3] = \mathbf{c}H$,	$\mathbf{c}R$,	$\mathbf{c}R^{-1} = \mathbf{c}RH$,
$\mathbf{c}V = \mathbf{c}VH$,	$\mathbf{c}RV$,	$\mathbf{c}R^{-1}V = \mathbf{c}RHV = \mathbf{c}VRH$.

Note that this set of vectors is rotated to itself by R and by R^{-1}, and reflected to itself by V and by H. Of course, scalar products of pairs of vectors are unchanged when both vectors are subjected to the same rotation or reflection. Therefore, instead of computing 36 scalar products

of all pairs from the set of nine, we need compute only 8 scalar products of **b** with all others, plus 7 of **c** with all others, to determine all 36 scalar products and hence all 36 distances.

The scalar products of **b** with its four nearest neighbors **c**R, **c**RV, **c**RH and **c**RHV are all 1/3 , as are the scalar products of **c** with its four nearest neighbors **b**R, **b**RH, **c**R and **c**RH . The scalar products of **b** with its other four neighbors **c**, **c**V, **b**R and **b**RH are all negative, as are the scalar products of **c** with its other four neighbors **b**, **c**V, **c**RV and **c**RHV .



The figure shows all nine points connected only to their nearest neighbors.