How The Randomizer Works - Methodology

For randomization, The Randomizer uses a proprietary sequence of random seed generations - it's not just 'pick one random number' and start a randomization sequence as most all other programs do. There are at least four sequences occurring:

1. Generate a random seed number.
2. Use random seed 1 to base the generation of seed 2.
3. Use seed 2 to generate 3, (this assures no patterns can develop).
4. In addition, there is a sequence based on the number of records in the dataset, and a sequence based on the characters from within the dataset assuring completely fair randomization.

The Randomizer initially uses a randomize function to return random single-precision numbers between 0.000000 and 1.000000. The initial random seed numbers are determined using the system timer as their base. Then the program reseeds (or initially starts) a given sequence returned by the first function. The Randomizer uses the linear-congruent method for random-number generation.

The following is an example of the linear-congruent method formula, similar to that used by The Randomizer:

```
x1 = ( x0 * a + c ) MOD 2^24
```

In the above example, the variables equal the following:
x1=new number
x0=previous number
a=214013
c=2531011

(Note: the MOD operator in the formula above returns the integer remainder after an integer division.)

The expression x1/(2^24) returns a floating-point number between 0.0 and 1.0.

Legal Implications of Methodology and Fairness

Lawsuits in Drug Testing Programs

In conducting drug testing, employers must balance legal liabilities due to lawsuits (brought by unhired applicants and employees who refuse to take the test or who are discharged or disciplined for positive test results) against the well-being of customers, clients, fellow employees, and members of the general public who may be injured or affected by a drug-using employee. Settlements in the former category are usually in the low thousands of dollars, while those in the latter are often in the millions.

Courts are holding more and more companies responsible for mistakes made by poorly trained personnel operating without well-conceived guidelines. As courts have declared**,** there is enormous liability when a company does nothing or does the wrong thing in the face of the clear evidence of drug and/or alcohol abuse throughout the workplaces of our country.

It is important that corporations and institutions disclose to those being tested that a random testing program is taking place.

In most all environments, agreement to participate in a testing program is a condition of employment or enrollment, and right or wrong, rights to bringing suit are waived with the acceptance of that employment or enrollment. So the legal implications are a) A professionally administered testing program is not in any significant danger of legal action from a test participant, and b) If legal action was to take place, an expert witness could dispel any reasonable doubt that the random selection process was somehow 'rigged' to produce specific results or was discriminatory.

In the event that there is some type of lawsuit by an individual brought against a testing program, rarely is the actual testing methodology as it applies to selection procedures relevant. Suits are typically of the types listed below (from Indiana Prevention Resource Center at Indiana University Drug Testing Prevention Primer which is attached):

“Drug testing related lawsuits filed against employers include invasion of privacy, wrongful discharge, defamation, intentional infliction of emotional distress, employer negligence, assault and battery, false imprisonment, and discrimination against minorities or people with disabilities.”

The point here is that the only type of suit mentioned above that is at all related to test selection is that of discrimination against minorities or people with disabilities. Since The Randomizer only requires Name and/or Social Security Number (or Employee Number) as data, discrimination is not possible if all employee data is entered.

Methodology Challenges and Statistical Implications

Statistically, when using a truly random method of selection, factorials are useful. They can **show how many different ways there are to order or arrange a set** of things. For example, if you have 5 books on a shelf, and want to know how many different ways there are to order or arrange them, simply find the factorial of 5:

5! = 5 x 4 x 3 x 2 x 1 = 120

This is relevant when using The Randomizer as well – using the above example if you had a population of only 5 employees, and continually ran random lists, theoretically, you could generate 120 lists of the 5 employees all arranged differently. So, in our theoretical example, if you randomized you list every month with the intent of testing the first person on the list the first week of the month, the second person on the second week, the third on the third, etc. you could achieve 120 different patterns. That’s not to say if you ran 120 lists they would all be different – but the example is useful because the combinations grow exponentially as the population of the list increases. So, if you have 100 employees, the number of different combinations is very large (example below), and the odds of patterns developing are very small. As with any random methodology, there is always a chance of the same person being placed in the same position more than once, but as the population on your employee list grows, the odds of this happening decrease. A useful example may be a slot-machine, or lotto where theoretically a person can win 2 or more times in a row, but it is a rareity.

Some examples of factorials are shown below – this can give a feel for the number of distinct possibilities from a population set based on the number of employees on a list – although there are no guarantees when using a random system, the numbers are useful and assumptions may be drawn to as how unlikely it is that patterns will develop as the list increases in size.

Here are some factorials:
1! = 1 = 1
2! = 2 x 1 = 2
3! = 3 x 2 x 1 = 6
4! = 4 x 3 x 2 x 1 = 24
5! = 5 x 4 x 3 x 2 x 1 = 120
6! = 6 x 5 x 4 x 3 x 2 x 1 = 720
7! = 7 x 6 x 5 x 4 x 3 x 2 x 1 = 5,040
8! = 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1 = 40,320
9! = 9 x 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1 = 362,880
10! = 10 x 9 x 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1 = 3,628,800

11! = 11 x 10 x 9 x 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1 = 39,916,800
12! = 12 x 11 x 10 x 9 x 8 x 7 x 6 x 5 x 4 x 3 x 2 x 1 = 479,001,600

100! =
93,326,215,443,944,152,681,699,238,856,266,700,490,715,968,264,381,621,468,592,963,895,217,599,993
229,915,608,941,463,976,156,518,286,253,697,920,827,223,758,251,185,210,916,864,000,000,000,000,
000,000,000,000

One may argue that if you are only selecting one name from your population list per interval for testing and if you have 100 names on the list, there are only 100 different possibilities and the chances of picking the same person every time a random list is run is 1 in 100. This is true with any random system, whether it is picking the names from a hat or flipping a coin. The reality is however, that generally more than one name is selected at a time – so if you are picking 5 names for testing from our example of 100 the chances that the same 5 names will be picked more than once are extremely small.

Because The Randomizer uses a proprietary sequence of random generation as outlined at the beginning of this document, true random selection can be achieved. In addition, because the data required for The Randomizer is name and number based only, discrimination cannot occur if all employees are entered in the same fashion.