Generatory liczb pseudolosowych oparłem o generator multiplikatywny, w którym a = 16807 oraz m = 2^31-1 = 0x7fffffff. Aby nie wyjść poza zakres 32-bitowego „inta” (po pomnożeniu przez a) generowane wartości zapisywałem, jako 64-bitowy *long long int*. Po podzieleniu uzyskanej liczby całkowitej przez m (i odpowiednim rzutowaniu na *double*) otrzymałem liczbę z zakresu (0, 1). Wyznaczyłem 10000 kolejnych wartości zgodnie z poniższym kodem.

```
std::ofstream myfile;
myfile.open("seeds.txt");
auto* generator = new Generators(1);
for(int i=0; i<10000; i++)
{
	const auto amount_from_rng = generator->uniform_01_distribution();
	myfile<<amount_from_rng<<"\n";
}
generator->~Generators();
myfile.close();
```

Następnie przekopiowałem dane z pliku tekstowego do Excela i sporządziłem histogram, który eksperymentalnie potwierdził, że powyższy generator generuje rozkład jednostajny z zakresu (0, 1). Wszystkie pozostałe generatory korzystają z tego rozkładu.

Aby zapewnić, że poszczególne zdarzenia będą korzystały z innego podciągu generowanych liczb, na początku wygenerowałem 20 ziaren (jąder) odległych od siebie o 100000 liczb. Każda z rodzajów generowanych wartości (czas przybycia dawcy, ilość potrzebnej krwi u pacjenta itp.) będzie zaczynać się od innego z tych ziaren. Ziarna zapisałem potem w tablicy w pliku *seeds.cpp*. Kod:

```
std::ofstream myfile;
myfile.open("seeds.txt");
auto* generator = new Generators(1);
for (int j = 0; j < 20; j++) {
	for (int i = 0; i < 100000; i++)
	{
		auto amount_from_rng = generator->uniform_01_distribution();
	}
	myfile << generator->get_seed()<<", ";
}
generator->~Generators();
myfile.close();
```

Pozostałe rozkłady uzyskałem w następujący sposób:
*U = rozkład równomierny (0,1)*

Rozkład równomierny <min, max> otrzymałem wyliczając:
U*(max+1-min) + min
Tę jedynkę dodałem aby zrekompensować rzutowanie na *int*.

Rozkład wykładniczy wyliczyłem korzystając z metody odwrotnej dystrybuanty:
-średnia * ln(U)

Rozkład geometryczny wyznaczyłem licząc, za którą próbą wylosowane U będzie mniejsze od 1/średnia.

Rozkład normalny N(0, 1) wyznaczyłem dodając 12 kolejnych U, a następnie odejmując 6. Skorzystałem tutaj z centralnego twierdzenia granicznego, wyznaczona suma jest dość dobrym przybliżeniem rozkładu normalnego.

Rozkład normalny (mi, sigma) wyliczamy:
N(0,1) * sigma + mi

Histogramy powyższych przykładów znajdują się w dołączonym arkuszu kalkulacyjnym.