Time	Event	Next Arrival / Departure Time (Client)	Queue(Remaining Time)
0	Start System	$1(A)$ / -	$\{\}$
1	A Arrives	$3(B)$ / $5(A)$	$\{A(4)\}$
3	B Arrives	$5(C)$ / $5(A)$	$\{A(2), B(3)\}$
5	A Completes, C Arrives	$7(D)$ / $6(C)$	$\{C(1), B(3)\}$
6	C Departs	$7(D)$ / $9(B)$	$\{B(3)\}$
7	D Arrives	$11(E)$ / $9(B)$	$\{B(2), D(8)\}$
9	B Completes	$11(E)$ / $17(D)$	$\{D(8)\}$
11	E Arrives	$15(F)$ / $17(D)$	$\{D(6), E(2)\}$
15	F Arrives	- / $17(D)$	$\{D(2), E(2), F(?)\}$

Table 1: Queue Schedule

Problem 1 (10 pts): Define/explain the following terms or concepts.

- Strong Law of Large Numbers
- Central Limit Theorem
- Confidence Interval
- Acceptance-Rejection RV generation

Problem 2 (10 pts): Consider a single-server queue in which customers are served according to a last in / first out non-preemptive discipline. When a given customer completes service, the server begins processing the most recently arrived customer. Customers arrive at times 1,3,5,7,11 and 15. With corresponding service requirements 4,3,1,8 and 2. How many customers are in the system at $t = 13.7$?

Solution: See Table 1. There are 2 customers in the system at time 13.7.

Problem 3 (10 pts):

1. A mean zero unit variance random variable X has a *Laplace* distribution if its p.d.f. is

$$f(x) = \frac{1}{2}e^{-|x|}.$$

Give an algorithm to generate such random variables.

Solution: It is easily shown that

$$F(x) = \begin{cases} \frac{1}{2}e^{x} & x \leq 0 \\ 1 - \frac{1}{2}e^{-x} & x > 0 \end{cases}, \quad \text{so} \quad F^{-1}(y) = \begin{cases} \ln 2y & 0 \leq y \leq \frac{1}{2}. \\ -\ln 2(y-1) & \frac{1}{2} < y \leq 1. \end{cases}$$

We now use the method of inversion, see Listing 1.

Listing 1: Inversion Algorithm

```
Generate a uniform (0,1) random variable u
Return X = F^{-1}(u)
```

2. Using the result above, give an algorithm to generate $\mathrm{N}\left(\mu, \sigma^2\right)$ random variables.

Solution: For this problem we will use the acceptance / rejection algorithm. Note that

$$\mathrm{N}\left(\mu, \sigma^2\right) \overset{\mathcal{D}}{=} \mu + \sigma\,\mathrm{N}\left(0, 1\right)$$

so we need only generate $\mathrm{N}(0,1)$ rv's and then rescale and center them. The *pdf* of a $\mathrm{N}(0,1)$ is

$$f_G(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}.$$

We wish to find a constant c such that

$$c = \sup\{f_G(x)/f_L(x) : x \in \mathbb{R}\}$$

for which it suffices to find a c such that

$$c \geq \frac{f_G(x)}{f_L(x)}.$$

Note the following

$$(1 - |x|)^2 \geq 0 \Rightarrow 1 - 2|x| + |x|^2 \geq 0 \quad \text{and thus} \quad \frac{1}{2} - |x| \geq -\frac{x^2}{2}.$$

It follows that

$$e^{\frac{1}{2}-|x|} \geq e^{-\frac{x^2}{2}} \Rightarrow 2\sqrt{e} f_L(x) \geq \sqrt{2\pi} f_G(x) \Rightarrow \frac{f_G(x)}{f_L(x)} \leq \sqrt{\frac{2e}{\pi}} = c.$$

(Note: this bound is tight, which is important as c^{-1} is the expected number of iterations in the acceptance rejection algorithm.) We use the following acceptance / rejection algorithm, see Listing 2.

Listing 2: Acceptance-Rejection Algorithm

```
Generate a Laplace random variable X (using method in part 1)
Generate an independent uniform (0,1) random variable U.
Is U ≤ f_G(Z)/(c · f_L(Z))?
          Yes: Return X = μ + σZ (''accept Z'')
          No:  Repeat  (''reject Z'')
```

Problem 4 (10 pts):

1. Suppose that we wish to compute $\alpha = \mathbf{E}g(W)$, where g is a non-negative function and W is a rv having density f_W. If X is a rv having positive density f_X, prove that α can be re-expressed as

$$\alpha = \mathbf{E}g(X)\frac{f_W(X)}{f_X(X)}$$

Solution:

$$\mathbf{E}g(W) = \int_{-\infty}^{\infty} g(w) f_W(w) dw = \int_{-\infty}^{\infty} g(x) f_W(x) \frac{f_X(x)}{f_X(x)} dx = \mathbf{E}\left[\frac{f_W(X)}{f_X(X)} g(X)\right].$$

2. Prove that the variance of the Monte Carlo procedure associated with the above is minimized by choosing X to have density

$$f_X^*(x) = g(x) f_W(x)/\alpha.$$

Solution:

$$\mathrm{Var}\left(g(W)\right) = \mathbf{E}\left[g(W)^2\right] - \mathbf{E}\left[g(W)\right]^2 = \mathbf{E}\left[g^2(X)\frac{f_W^2(X)}{f_{X^*}^2(X)}\right] - \alpha^2 = \mathbf{E}\left[\alpha^2\right] - \alpha^2 = 0$$

which is as minimal as variance can be.

Prof. Peter Glynn (glynn@stanford.edu)
Stanford University

3. Why is the above choice of $f_X^*(\cdot)$ impractical in general?

   Solution: It relies on knowing a priori α.

4. Suppose that we wish to compute $\alpha = \mathbf{P}\{N(0,1) > 3.75\}$. Compute α first by (crude) Monte Carlo sampling based on iid sampling of $\mathbb{1}_{\{N(2,1)>5.75\}}$, and then compute α via the approach suggested in 3. with density

$$f_X(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-(x-5.75)^2/2\right)$$

   (i.e. sampling X according to a $N(5.75,1)$ distribution). Produce a 90% confidence interval based on 10000 and 100000 samples for each approach.

   Remark. The Monte Carlo approach described above is called "importance sampling".

   Solution: The following MATLAB code was use to compute the "crude" confidence intervals (see 3).

Listing 3: Monte Carlo Simulation

```
clear all;

N = 10000;

norms = zeros(N,1);
g = norms;

h = g;

s = 0;

for i = 1:N
    norms(i) = randn;
end

h = (norms >= 3.75);

g = (norms >= 0) .* exp(-1/2 * ((norms +3.75).^2 - (norms).^2));

pc = sum(h) / N

p = sum(g) / N;
p

s = sqrt(1/(N-1) * sum( (g - p) .* (g-p)));

lb = p - 1.645 * s / sqrt(N)
ub = p + 1.645 * s / sqrt(N)

1 - normcdf(5.75, 2,1)
```

   The value stored in `pc` is the crude probability. It typically is 0, and thus computing a confidence interval makes no sense in this context. At 100,000 samples, results are similarly poor. One run produce the following results for the importance sampling methods (values stored in `p`, `lb` and `ub`)

$$\hat{\alpha} = 8.8493 \times 10^{-5} \qquad \alpha \in [8.7548, 8.9439] \times 10^{-5}.$$