Solution to the MMF 2021 Midterm Test for 2019

1. [10 marks: 5 marks for each part]

   I asked the students to consider the expression

   $$\sqrt{1+x} - \sqrt{1-x} \tag{1}$$

   for $x \in [-1, 1]$.

   (a) I asked them for what range of values of x does expression (1) produce inaccurate results in IEEE double-precision floating-point arithmetic?
   (By inaccurate I mean that the absolute value of the relative error in the floating-point approximation to (1) is orders of magnitude larger than *machine epsilon* for this floating-point number system.)
   I asked them to justify their answer.

   The loss of accuracy in (1) arises from catastrophic cancellation in the subtraction between $\sqrt{1+x}$ and $\sqrt{1-x}$ when $|x|$ is small, but nonzero, in which case both $\sqrt{1+x} \approx 1$ and $\sqrt{1-x} \approx 1$.
   For example, suppose x is a nonzero, normalized IEEE floating-point number satisfying $|x| < \frac{1}{4}\,\epsilon_{\text{mach}}$. Then both $\text{fl}(1+x) = 1$ and $\text{fl}(1-x) = 1$. Consequently, $\text{fl}(\sqrt{1+x}) = 1$ and $\text{fl}(\sqrt{1-x}) = 1$. Therefore,

   $$A = \text{fl}(\sqrt{1+x} - \sqrt{1-x}) = 0$$

   However, since $x \neq 0$, the true value of (1) is not 0. That is,

   $$T = \sqrt{1+x} - \sqrt{1-x} \neq 0$$

   Therefore, the relative error in the computation of (1) is

   $$\frac{A-T}{T} = -1$$

   So, in this case, the absolute value of the relative error in the floating-point approximation to (1) is orders of magnitude larger than *machine epsilon*, which for IEEE double-precision floating-point arithmetic is about $2.22 \cdot 10^{-16}$. That is, the computed value of (1) is extremely inaccurate in a relative error sense.

   Note that it is important that they exclude $x = 0$, since, for $x = 0$, both the exact and computed values for (1) are 0. Although, the relative error in this case is technically $0/0$, it seems reasonable to extend the definition in this case to say that the relative error is 0, since the absolute error is 0.

   Some students might think the computation of (1) will be inaccurate if $x \approx 1$ or $x \approx -1$, since in this case there will be cancellation in the computation of either $1 - x$ or $1 + x$, respectively. However, this will not cause a serious loss of accuracy in the computation of (1).

   The students don't have to give as extreme an example as I have above, but they should show that there is a nonzero x, with $|x|$ small, for which the absolute value of the relative error in the floating-point approximation to (1) is orders of magnitude larger than *machine epsilon*.

   Marking: give them 3 marks for identifying that the computed value of (1) will be very inaccurate for a nonzero x satisfying $|x| \ll 1$. Given them an additional 2 marks for correctly justifying their answer.
   Of course, you can give them part marks for each of the points above if their answer is on the right track, but not completely correct.

(b) I asked the students to give another expression that is mathematically equal to (1), but is computationally much more effective than (1) in the sense that it is accurate in IEEE double-precision floating-point arithmetic for all $x \in [-1, 1]$.
(By accurate I mean that the the absolute value of the relative error associated with your new floating-point expression is less than 10 times *machine epsilon*.)
I also asked them to justify their answer.

One way to find another expression that is mathematically equal to (1), but is computationally much more effective than (1), is to multiply (1) by what I sometimes call the *conjugate* of (1) as follows.

$$\begin{aligned}\sqrt{1+x} - \sqrt{1-x} &= \left(\sqrt{1+x} - \sqrt{1-x}\right) \frac{\sqrt{1+x} + \sqrt{1-x}}{\sqrt{1+x} + \sqrt{1-x}} \\ &= \frac{(1+x) - (1-x)}{\sqrt{1+x} + \sqrt{1-x}} \\ &= \frac{2x}{\sqrt{1+x} + \sqrt{1-x}}\end{aligned}$$

We show below that

$$\frac{2x}{\sqrt{1+x} + \sqrt{1-x}} \tag{2}$$

is computationally much more effective than (1) in the sense that it is accurate in IEEE double-precision floating-point arithmetic for all $x \in [-1, 1]$.
To see that (2) is accurate in IEEE double-precision floating-point arithmetic for all $x \in [-1, 1]$, consider

$$\begin{aligned}&\text{fl}\left(\frac{2x}{\sqrt{1+x} + \sqrt{1-x}}\right) \\ &\quad = \frac{2x(1+\delta_1)}{\Big(\sqrt{(1+x)(1+\delta_2)}(1+\delta_3) + \sqrt{(1-x)(1+\delta_4)}(1+\delta_5)\Big)(1+\delta_6)}\,(1+\delta_7)\end{aligned} \tag{3}$$

where $|\delta_i| \leq \frac{1}{2}\,\epsilon_{\text{mach}}$ for $i = 1, 2, \ldots, 7$.
First note that

$$\begin{aligned}\sqrt{(1+x)(1+\delta_2)} &= \sqrt{1+x}\,\sqrt{1+\delta_2} \\ &= \sqrt{1+x}\,(1+\hat{\delta}_2)\end{aligned} \tag{4}$$

for some $\hat{\delta}_2$ satisfying $|\hat{\delta}_2| \leq \frac{1}{2}\,\epsilon_{\text{mach}}$, and

$$\begin{aligned}\sqrt{(1-x)(1+\delta_4)} &= \sqrt{1-x}\,\sqrt{1+\delta_4} \\ &= \sqrt{1-x}\,(1+\hat{\delta}_4)\end{aligned} \tag{5}$$

for some $\hat{\delta}_4$ satisfying $|\hat{\delta}_4| \leq \frac{1}{2}\,\epsilon_{\text{mach}}$. I explained (4) and (5) to them in class. So, it is fine if they just use them here without proof.
Substitute (4) and (5) into (3) to get

$$\begin{aligned}&\text{fl}\left(\frac{2x}{\sqrt{1+x} + \sqrt{1-x}}\right) \\ &\quad = \frac{2x(1+\delta_1)}{\Big(\sqrt{1+x}\,(1+\hat{\delta}_2)(1+\delta_3) + \sqrt{1-x}\,(1+\hat{\delta}_4)(1+\delta_5)\Big)(1+\delta_6)}\,(1+\delta_7)\end{aligned} \tag{6}$$

Since both $\sqrt{1+x} \geq 0$ and $\sqrt{1-x} \geq 0$ for all $x \in [-1,1]$, it follows that

$$\begin{aligned} &\sqrt{1+x}\,(1+\hat{\delta}_2)(1+\delta_3) + \sqrt{1-x}\,(1+\hat{\delta}_4)(1+\delta_5) \\ &= \Big(\sqrt{1+x}+\sqrt{1-x}\Big)(1+\tilde{\delta}_2)(1+\tilde{\delta}_3) \end{aligned} \tag{7}$$

for some $\tilde{\delta}_2$ and $\tilde{\delta}_3$ satisfying $|\tilde{\delta}_2| \leq \frac{1}{2}\,\epsilon_{\text{mach}}$ and $|\tilde{\delta}_3| \leq \frac{1}{2}\,\epsilon_{\text{mach}}$. I explained (7) in class. So, it is fine if they just use it here without proof.
Substitute (7) into (6) to get

$$\begin{aligned} &\text{fl}\left(\frac{2x}{\sqrt{1+x}+\sqrt{1-x}}\right) \\ &\quad = \frac{2x(1+\delta_1)}{\Big(\sqrt{1+x}+\sqrt{1-x}\Big)(1+\tilde{\delta}_2)(1+\tilde{\delta}_3)(1+\delta_6)}\,(1+\delta_7) \\ &\quad = \frac{2x}{\sqrt{1+x}+\sqrt{1-x}}\,\frac{(1+\delta_1)(1+\delta_7)}{(1+\tilde{\delta}_2)(1+\tilde{\delta}_3)(1+\delta_6)} \end{aligned} \tag{8}$$

I told them in class, that, if $|\delta| \leq \frac{1}{2}\,\epsilon_{\text{mach}}$ and $\epsilon_{\text{mach}} \ll 1$, then

$$\frac{1}{1+\delta} = 1+\check{\delta} \tag{9}$$

for some $\check{\delta}$ satisfying $|\check{\delta}| \leq \frac{1.01}{2}\,\epsilon_{\text{mach}}$. So, using (9) in (8) three times, we get

$$\begin{aligned} &\text{fl}\left(\frac{2x}{\sqrt{1+x}+\sqrt{1-x}}\right) \\ &\quad = \frac{2x}{\sqrt{1+x}+\sqrt{1-x}}\,(1+\delta_1)(1+\delta_7)(1+\check{\delta}_2)(1+\check{\delta}_3)(1+\check{\delta}_6) \end{aligned} \tag{10}$$

Now note that

$$\begin{aligned} &(1+\delta_1)(1+\delta_7)(1+\check{\delta}_2)(1+\check{\delta}_3)(1+\check{\delta}_6) \\ &\quad = 1+\delta_1+\delta_7+\check{\delta}_2+\check{\delta}_3+\check{\delta}_6+\text{h.o.t.} \\ &\quad = 1+\delta \end{aligned} \tag{11}$$

where

$$\delta = \delta_1+\delta_7+\check{\delta}_2+\check{\delta}_3+\check{\delta}_6+\text{h.o.t}$$

and h.o.t. stands for *higher order terms*. These are terms such as $\delta_1\delta_7$, $\delta_1\check{\delta}_2$, etc. Note that

$$\begin{aligned} |\delta| &= |\delta_1+\delta_7+\check{\delta}_2+\check{\delta}_3+\check{\delta}_6+\text{h.o.t}| \\ &\leq |\delta_1|+|\delta_7|+|\check{\delta}_2|+|\check{\delta}_3|+|\check{\delta}_6|+|\text{h.o.t}| \end{aligned} \tag{12}$$

Since $\delta_i \leq \frac{1}{2}\,\epsilon_{\text{mach}}$ for $i=1,7$ and $\check{\delta}_i \leq \frac{1.01}{2}\,\epsilon_{\text{mach}}$ for $i=2,3,6$, it follows that

$$|\delta_1|+|\delta_7|+|\check{\delta}_2|+|\check{\delta}_3|+|\check{\delta}_6| \leq \frac{5.03}{2}\,\epsilon_{\text{mach}} \tag{13}$$

I also told them in class that, if $\epsilon_{\text{mach}} \ll 1$, they can bound the h.o.t by

$$|\text{h.o.t}| \leq \frac{1}{2}\,\epsilon_{\text{mach}} \tag{14}$$

(Actually, since $\epsilon_{\text{mach}} = 2.22 \cdot 10^{-16}$, you can get a much tighter bound than (14), but (14) is fine for our purposes.) Therefore, using (12), (13) and (14), we get

$$|\delta| \leq \frac{6.03}{2}\,\epsilon_{\text{mach}} = 3.015\,\epsilon_{\text{mach}}$$

Hence, we have shown that

$$\text{fl}\left(\frac{2x}{\sqrt{1+x}+\sqrt{1-x}}\right) = \frac{2x}{\sqrt{1+x}+\sqrt{1-x}}\,(1+\delta)$$

for some δ satisfying $|\delta| \leq 3.015\,\epsilon_{\text{mach}}$. Since δ is the relative error in the computation of

$$\frac{2x}{\sqrt{1+x}+\sqrt{1-x}}$$

we have shown that (2) is accurate (in a relative error sense) in IEEE double-precision floating-point arithmetic for all $x \in [-1, 1]$.

It's fine is they get a bound on the relative error that is less tight than the one I have developed above. Recall that I asked them to show only that the relative error associated with their new floating-point expression is less than 10 times *machine epsilon*.)

Marking: give them 2 marks for deriving (2) and 3 marks for explaining why it is accurate.
Of course, you can give them part marks for each of the points above if their answer is on the right track, but not completely correct.

Some students might give a "hand-wavy" argument for why (2) is accurate. Given them 0, 1 or 2 (usually 1 or 2) for this, depending on how believable you find their argument.

2. [5 marks]

The proof that acceptance-rejection method given in this question for a discrete random variable is very similar to the proof I gave in class for the acceptance-rejection method for continuous random variable.

For the random variable X generated by the method given in this question,

$$\begin{aligned}\mathbb{P}(X=k) &= \mathbb{P}\left(Y=k \,\middle|\, U \le \frac{p(Y)}{c\,q(Y)}\right) \\ &= \frac{\mathbb{P}\left(Y=k \;\&\; U \le \frac{p(Y)}{c\,q(Y)}\right)}{\mathbb{P}\left(U \le \frac{p(Y)}{c\,q(Y)}\right)} \end{aligned} \tag{15}$$

Note that

$$\begin{aligned}\mathbb{P}\left(U \le \frac{p(Y)}{c\,q(Y)}\right) &= \sum_{k=0}^{\infty} \mathbb{P}\left(U \le \frac{p(Y)}{c\,q(Y)} \,\middle|\, Y=k\right) \mathbb{P}\left(Y=k\right) \\ &= \sum_{k=0}^{\infty} \mathbb{P}\left(U \le \frac{p(k)}{c\,q(k)}\right) \mathbb{P}\left(Y=k\right) \\ &= \sum_{k=0}^{\infty} \frac{p(k)}{c\,q(k)} q(k) \qquad (\text{since } U \sim \text{Unif}[0,1]) \\ &= \frac{1}{c} \sum_{k=0}^{\infty} p(k) \\ &= \frac{1}{c} \end{aligned} \tag{16}$$

Combining (15) and (16), we get

$$\begin{aligned}\mathbb{P}(X=k) &= \frac{\mathbb{P}\left(Y=k \;\&\; U \le \frac{p(Y)}{c\,q(Y)}\right)}{1/c} \\ &= c\,\mathbb{P}\left(Y=k \;\&\; U \le \frac{p(Y)}{c\,q(Y)}\right) \\ &= c\,\mathbb{P}\left(Y=k \;\&\; U \le \frac{p(k)}{c\,q(k)}\right) \\ &= c\,\mathbb{P}(Y=k)\,\mathbb{P}\left(U \le \frac{p(k)}{c\,q(k)}\right) && (\text{since } Y \text{ and } U \text{ are independent}) \\ &= c\,q(k)\,\frac{p(k)}{c\,q(k)} && (\text{since } U \sim \text{Unif}[0,1]) \\ &= p(k) \end{aligned}$$

Therefore, we have shown $\mathbb{P}(X=k) = p(k)$ as required.

Marking: use the usual marking scheme:

- 5 marks if essentially everything is correct,
- 4 marks if most everything is correct, but there are a few minor errors,

- 3 marks if they are clearly on the right path, but there are some significant errors,
- 2 marks if they have a good start, but it is far from a correct answer,
- 1 mark if there is anything useful in their answer,
- 0 marks if there is nothing useful in their answer.

3. [5 marks]

Our goal is to find a method that generates a random variable, X, with probability density function (pdf) $g(x)$ that requires only one Unif$[0,1]$ random variable for each X that it generates. The key to finding such a method is to use the inverse transform method with the cumulative distribution function (CDF) $G(x)$ associated with the probability density function (pdf) $g(x)$.

Recall that the pdf $g(x)$ for this question is defined as follows. If $k \in \{1, 2, \ldots, L\}$ and $x \in [(k-1)/L, k/L)$, then

$$g(x) = Lq_k$$

where, in addition, we set $g(1) = Lq_L$. It is noted in the question that

$$q_k > 0$$

and

$$\sum_{k=1}^{L} q_k = 1$$

See the question for a confirmation of this.

So, the CDF $G(x)$ associated with the pdf $g(x)$ is

$$G(x) = \int_0^x g(t)\,dt$$

Note that, if $x \in [(k-1)/L, k/L)$, then

$$\begin{aligned} G(x) &= \int_0^x g(t)\,dt \\ &= \int_0^{(k-1)/L} g(t)\,dt + \int_{(k-1)/L}^{x} g(t)\,dt \\ &= \sum_{j=1}^{k-1} \int_{(j-1)/L}^{j/L} g(t)\,dt + \int_{(k-1)/L}^{x} g(t)\,dt \\ &= \sum_{j=1}^{k-1} \int_{(j-1)/L}^{j/L} Lq_j\,dt + \int_{(k-1)/L}^{x} Lq_k\,dt \\ &= \sum_{j=1}^{k-1} Lq_j\,\frac{1}{L} + Lq_k(x - (k-1)/L) \\ &= \sum_{j=1}^{k-1} q_j + q_k(Lx - (k-1)) \end{aligned} \tag{17}$$

and

$$\begin{aligned} G(1) &= \int_0^1 g(t)\,dt \\ &= \sum_{j=1}^{L} q_j \\ &= 1 \end{aligned}$$

Note that, in (17) above, if $k = 1$, we take

$$\sum_{j=1}^{k-1} q_j = \sum_{j=1}^{0} q_j = 0$$

Now the inverse transform method works as follows.

(i) Generate a $U \sim \text{Unif}[0, 1]$

(ii) Set $X = G^{-1}(U)$

We need to expand a little on step (ii) above.

First note that $X = G^{-1}(U)$ is equivalent to $G(X) = U$. So, given U from step (i), we need to solve $G(X) = U$ for X in step (ii). To this end, note that if

$$\sum_{j=1}^{k-1} q_j \leq U < \sum_{j=1}^{k} q_j$$

then there is an $X \in [(k-1)/L, k/L)$ such that

$$G(X) = U$$

since

$$G((k-1)/L) = \sum_{j=1}^{k-1} q_j \leq U$$

$$G(k/L) = \sum_{j=1}^{k} q_j > U$$

and $G(X)$ is continuous.

To solve $G(X) = U$, given that $X \in [(k-1)/L, k/L)$, note

$$G(X) = \sum_{j=1}^{k-1} q_j + q_k(LX - (k-1)) = U$$

is equivalent to

$$q_k(LX - (k-1)) = U - \sum_{j=1}^{k-1} q_j$$

whence

$$X = \frac{k-1}{L} + \frac{1}{Lq_k}\left(U - \sum_{j=1}^{k-1} q_j\right)$$

So we can rewrite that inverse transform method above as follows.

(i) Generate a $U \sim \text{Unif}[0,1]$

(iia) Find $k \in \{1, 2, \ldots, L\}$ such that

$$\sum_{j=1}^{k-1} q_j \leq U < \sum_{j=1}^{k} q_j$$

If $U = 1$, set $k = L$.

(iib) Set

$$X = \frac{k-1}{L} + \frac{1}{Lq_k}\left(U - \sum_{j=1}^{k-1} q_j\right)$$

Note that the formula for X in Step (iib) works correctly for $U = 1$, since in this case, $k = L$ from step (iia) and so

$$\begin{aligned} X &= \frac{L-1}{L} + \frac{1}{Lq_L}\left(U - \sum_{j=1}^{L-1} q_j\right) \\ &= \frac{L-1}{L} + \frac{1}{Lq_L}\left(1 - \sum_{j=1}^{L-1} q_j\right) \\ &= \frac{L-1}{L} + \frac{1}{Lq_L}\left(q_L\right) \\ &= \frac{L-1}{L} + \frac{1}{L} \\ &= 1 \end{aligned}$$

Marking: give them 2 marks if they realize that they should use the inverse transform method and another 3 marks if they develop a method similar to my method at the top of this page.

Of course, you can give them part marks for each of the points above if their answer is on the right track, but not completely correct.

4. [15 marks: 5 marks for each part]

First I'll repeat the question. My answer begins on page 12.

In many applications, you need to compute tail probabilities, $\mathbb{P}(X \geq x)$. For example, if $\mathcal{L}$ is the loss associated with defaults in a portfolio of bonds over some time period T, you might want to compute $\mathbb{P}(\mathcal{L} \geq l)$. This is required, for example, if you want to compute the Value-at-Risk associated with the portfolio.

However, to keep things simple in this question, we'll focus on computing $\mathbb{P}(X \geq 10)$, where $X \sim N(0,1)$ is a standard normal random variable (i.e., X is a normal random variable with mean 0 and variance 1). You might think that you can compute $\mathbb{P}(X \geq 10)$ from the CDF, $\Phi(x)$, of the normal distribution, since $\mathbb{P}(X \geq 10) = 1 - \Phi(10)$. Although this is true in theory, if you evaluate this in MatLab, you'll find that the computed value of $\Phi(10)$ is 1, whence the computed value of $1 - \Phi(10)$ is 0. So, this does not lead to a good approximation to $\mathbb{P}(X \geq 10)$. The reason for this is that $\mathbb{P}(X \geq 10) \approx 10^{-23}$. So, your computer would need to carry the equivalent of at least 23 decimal digits to be able to differentiate $\Phi(10)$ from 1.

Even if we could compute $\Phi(10)$ to sufficient accuracy, this would not help in the more realistic example $\mathbb{P}(\mathcal{L} \geq l)$ mentioned above. So, let's forget about computing $\mathbb{P}(X \geq 10)$ from $\Phi(x)$ for now.

Another approach is to approximate $\mathbb{P}(X \geq 10)$ by a Monte Carlo simulation. To this end, note that

$$\mathbb{P}(X \geq 10) = \int_{10}^{\infty} f(x)\,dx = \int_{-\infty}^{\infty} H_{10}(x) f(x)\,dx = \mathbb{E}_f[H_{10}(X)]$$

where $X \sim N(0,1)$,

$$f(x) = \frac{\mathrm{e}^{-x^2/2}}{\sqrt{2\pi}}$$

and

$$H_{10}(x) = \begin{cases} 1 & \text{if } x \geq 10 \\ 0 & \text{if } x < 10 \end{cases}$$

Hence, we can write a very simple Monte Carlo simulation to approximate $p = \mathbb{P}(X \geq 10) = \mathbb{E}_f[H_{10}(X)]$:

$$\hat{p} = \frac{1}{N} \sum_{i=1}^{N} H_{10}(X_i) \tag{18}$$

where $X_i \sim N(0,1)$. Suppose we want $\hat{p}$ to approximate $p = \mathbb{P}(X \geq 10)$ to at least two significance digits with a 95% confidence level.

(a) First show that the variance of $H_{10}(X)$ satisfies

$$\mathrm{Var}_f[H_{10}(X)] = \mathbb{E}_f[(H_{10}(X) - p)^2] = p - p^2 \tag{19}$$

Then use $\mathrm{Var}_f[H_{10}(X)]$ (even if you were not able to verify (19)) to estimate how large you need to choose N in (18) to achieve this level of accuracy.

Your estimation of N needs to be of the right order of magnitude only. So, you can use $z_{\delta/2} \approx 2$ for the 95% confidence level and $p \approx 10^{-23}$ in computing your estimate of N.

Your value of N computed in part (a) above should be so large that the Monte Carlo simulation (18) is completely impractical. However, we can use importance sampling to get a much more efficient Monte Carlo simulation. To this end, let

$$g(x) = \frac{\mathrm{e}^{-(x-10)^2/2}}{\sqrt{2\pi}}$$

be the probability density function for $Y \sim N(10,1)$. Then

$$\begin{aligned} p = \mathbb{P}(X \geq 10) &= \int_{10}^{\infty} f(x)\,dx \\ &= \int_{-\infty}^{\infty} H_{10}(x) f(x)\,dx \\ &= \int_{-\infty}^{\infty} H_{10}(x) \frac{f(x)}{g(x)} g(x)\,dx \\ &= \mathbb{E}_g\left[H_{10}(Y)\frac{f(Y)}{g(Y)}\right] \end{aligned} \tag{20}$$

where $X \sim N(0,1)$ and $Y \sim N(10,1)$.

(b) What is the Monte-Carlo importance-sampling simulation associated with $\mathbb{E}_g[H_{10}(Y)\frac{f(Y)}{g(Y)}]$ in (20) above to approximate $p = \mathbb{P}(X \geq 10)$?
Clearly state what random variables you are using in this Monte Carlo simulation and how you would compute them if you have a function such as MatLab's randn that returns $N(0,1)$ normal random variables.

Let

$$\mathrm{Var}_g\left[H_{10}(Y)\frac{f(Y)}{g(Y)}\right] = \mathbb{E}_g\left[\left(H_{10}(Y)\frac{f(Y)}{g(Y)} - p\right)^2\right]$$

where $Y \sim N(10,1)$. To assess the efficiency of the Monte-Carlo importance-sampling simulation in part (b) above compared to the simple Monte-Carlo simulation (18), we need to estimate how much smaller the variance $\mathrm{Var}_g\left[H_{10}(Y)\frac{f(Y)}{g(Y)}\right]$ is than the variance $\mathrm{Var}_f[H_{10}(X)]$.

It does not seem too easy to get a closed form expression for $\mathrm{Var}_g\left[H_{10}(Y)\frac{f(Y)}{g(Y)}\right]$, but it is not too hard to show

$$\mathrm{Var}_g\left[H_{10}(Y)\frac{f(Y)}{g(Y)}\right] \leq \mathrm{e}^{-50}p - p^2 \tag{21}$$

Thus, $\mathrm{Var}_g\left[H_{10}(Y)\frac{f(Y)}{g(Y)}\right]$ is about $\mathrm{e}^{-50} \approx 2 \times 10^{-22}$ times smaller than $\mathrm{Var}_f[H_{10}(X)]$. This is quite a significant variance reduction!

(c) Show that (21) is true.
Then use (21) (even if you were not able to prove it is true) to estimate how large you need to choose N in your Monte-Carlo importance-sampling simulation to achieve the same level of accuracy as was specified in part (a) above.
Your estimation of N needs to be of the right order of magnitude only. So, you can use $z_{\delta/2} \approx 2$ for the 95% confidence level, $p \approx 10^{-23}$ and $\mathrm{e}^{-50} \approx 2 \times 10^{-22}$ in computing your estimate of N.

My answer begins here.

(a) Since $p = \mathbb{E}_f[H_{10}(X)]$,

$$\begin{aligned}\mathrm{Var}_f[H_{10}(X)] &= \mathbb{E}_f[(H_{10}(X) - \mathbb{E}_f[H_{10}(X)])^2] \\ &= \mathbb{E}_f[(H_{10}(X) - p)^2] \\ &= \mathbb{E}_f[(H_{10}(X))^2] - p^2 \\ &= \mathbb{E}_f[H_{10}(X)] - p^2 \\ &= p - p^2\end{aligned}$$

where I used $(H_{10}(X))^2 = H_{10}(X)$, since $H_{10}(X)$ is either 0 or 1.
The 95% confidence interval for p is

$$\left(\hat{p} - \frac{z_{\delta/2}\,\sigma_f}{\sqrt{N}},\, \hat{p} + \frac{z_{\delta/2}\,\sigma_f}{\sqrt{N}}\right)$$

where $\sigma_f^2 = \mathrm{Var}_f[H_{10}(X)] = p - p^2$. So, if we want $\hat{p}$ to approximate p to at least two significance digits with a 95% confidence level, we need to choose N such that

$$\frac{z_{\delta/2}\,\sigma_f}{\sqrt{N}} \lesssim 10^{-2}\hat{p} \tag{22}$$

Since we want to choose N as small as possible, we should choose N to satisfy

$$\frac{z_{\delta/2}\,\sigma_f}{\sqrt{N}} \approx 10^{-2}\hat{p} \tag{23}$$

That is,

$$N \approx \frac{10^4\, z_{\delta/2}^2\, \sigma_f^2}{\hat{p}^2}$$

Recall that $z_{\delta/2} \approx 2$, $\hat{p} \approx p$, $p \approx 10^{-23}$ and $\sigma_f^2 = \mathrm{Var}_f[H_{10}(X)] = p - p^2$. So, $\sigma_f^2 \approx p$. Therefore,

$$N \approx \frac{10^4 \cdot 4 \cdot p}{p^2} = \frac{4 \cdot 10^4}{p} \approx 4 \cdot 10^4 \cdot 10^{23} = 4 \cdot 10^{27}$$

Marking:

- give them 2 marks for calculating $\mathrm{Var}_f[H_{10}(X)]$ correctly,
- give them 2 marks for realizing that, if they want to approximate p to at least two significance digits with a 95% confidence level, they need a bound on N like (22) (i.e., they need p or $\hat{p}$ on the right side of (22)),
- give them 1 mark for using some "reasonable" confidence interval to compute an approximation to N, even if they missed the p or $\hat{p}$ on the right side of (22).

Of course, you can give them part marks for each of the points above if their answer is on the right track, but not completely correct.

(b) The Monte-Carlo importance-sampling simulation associated with $\mathbb{E}_g[H_{10}(Y)\frac{f(Y)}{g(Y)}]$ in (20) to approximate $p = \mathbb{P}(X \geq 10)$ is

$$\hat{p} = \frac{1}{N}\sum_{i=1}^{N} H_{10}(Y_i)\frac{f(Y_i)}{g(Y_i)} \tag{24}$$

where $Y_i \sim N(10,1)$ and the Y_i, $i = 1,2,\ldots,N$, are independent. We noted in class that, to generate the independent $Y_i \sim N(10,1)$, for $i = 1,2,\ldots,N$, we can first generate independent $X_i \sim N(0,1)$, for $i = 1,2,\ldots,N$, and then set

$$Y_i = 10 + X_i, \qquad \text{for } i = 1,2,\ldots,N$$

Note that we can use randn to generate the independent $X_i \sim N(0,1)$ for $i = 1,2,\ldots,N$.
They could leave $\hat{p}$ as written above in (24) or they could simplify it a little further by noting that, for the pdfs $f(x)$ and $g(x)$ given above,

$$\begin{aligned} H_{10}(Y_i)\frac{f(Y_i)}{g(Y_i)} &= H_{10}(Y_i)\frac{\mathrm{e}^{-Y_i^2/2}}{\mathrm{e}^{-(Y_i-10)^2/2}} \\ &= H_{10}(10+X_i)\frac{\mathrm{e}^{-(X_i+10)^2/2}}{\mathrm{e}^{-X_i^2/2}} \\ &= H_{10}(10+X_i)\,\mathrm{e}^{-10X_i-50} \end{aligned}$$

Therefore, the $\hat{p}$ above in (24) can be rewritten as

$$\hat{p} = \frac{1}{N}\sum_{i=1}^{N} H_{10}(10+X_i)\,\mathrm{e}^{-10X_i-50} \tag{25}$$

where the independent $X_i \sim N(0,1)$, for $i = 1,2,\ldots,N$, can be computed by randn.

Marking:

- give them 3 marks for a Monte-Carlo importance-sampling simulation such as (24) or an equivalent one such as (25),
- give them 2 marks for explaining how to generate independent $Y_i \sim N(10,1)$, for $i = 1,2,\ldots,N$.

Of course, you can give them part marks for each of the points above if their answer is on the right track, but not completely correct.

(c) To prove that (21) is true, first recall that

$$p = \mathbb{E}_f[H_{10}(X)] = \mathbb{E}_g\left[H_{10}(Y)\frac{f(Y)}{g(Y)}\right]$$

Therefore,

$$\begin{aligned}
\mathrm{Var}_g\left[H_{10}(Y)\frac{f(Y)}{g(Y)}\right] &= \mathbb{E}_g\left[\left(H_{10}(Y)\frac{f(Y)}{g(Y)} - \mathbb{E}_g\left[H_{10}(Y)\frac{f(Y)}{g(Y)}\right]\right)^2\right] \\
&= \mathbb{E}_g\left[\left(H_{10}(Y)\frac{f(Y)}{g(Y)} - p\right)^2\right] \\
&= \mathbb{E}_g\left[\left(H_{10}(Y)\frac{f(Y)}{g(Y)}\right)^2\right] - p^2 \\
&= \int_{-\infty}^{\infty}\left(H_{10}(y)\frac{f(y)}{g(y)}\right)^2 g(y)\,dy - p^2 \\
&= \int_{-\infty}^{\infty}\left(H_{10}(y)\right)^2\frac{f(y)}{g(y)}f(y)\,dy - p^2 \\
&= \int_{-\infty}^{\infty}H_{10}(y)\frac{f(y)}{g(y)}f(y)\,dy - p^2 \\
&= \int_{10}^{\infty}\frac{f(y)}{g(y)}f(y)\,dy - p^2 \\
&= \int_{10}^{\infty}\mathrm{e}^{(-y^2/2+(y-10)^2/2)}f(y)\,dy - p^2 \\
&= \int_{10}^{\infty}\mathrm{e}^{(-10y+50)}f(y)\,dy - p^2 \\
&\leq \int_{10}^{\infty}\mathrm{e}^{-50}f(y)\,dy - p^2 \\
&= \mathrm{e}^{-50}\int_{10}^{\infty}f(y)\,dy - p^2 \\
&= \mathrm{e}^{-50}p - p^2
\end{aligned}$$

Therefore,

$$\mathrm{Var}_g\left[H_{10}(Y)\frac{f(Y)}{g(Y)}\right] \leq \mathrm{e}^{-50}p - p^2$$

whence (21) is true.
Let

$$\sigma_g^2 = \mathrm{Var}_g\left[H_{10}(Y)\frac{f(Y)}{g(Y)}\right]$$

and note that $\sigma_g^2 \leq \mathrm{e}^{-50}p - p^2 \approx \mathrm{e}^{-50}p$, since $p \approx 10^{-23}$ and $\mathrm{e}^{-50} \approx 2 \times 10^{-22}$. (Note that the approximation, $\mathrm{e}^{-50}p - p^2 \approx \mathrm{e}^{-50}p$, used here is not as tight as the ones used above, but it is still "good enough" for our purposes.)

Following a similar argument as in part (a), we get that

$$\begin{aligned} N &\approx \frac{10^4\, z_{\delta/2}^2\, \sigma_g^2}{\hat{p}^2} \\ &\lesssim \frac{10^4\, z_{\delta/2}^2\, \mathrm{e}^{-50}\, p}{p^2} \\ &\approx \frac{10^4 \times 4 \times \mathrm{e}^{-50}}{p} \\ &\approx \frac{10^4 \times 4 \times 2 \times 10^{-22}}{10^{-23}} \\ &\approx 8 \times 10^5 \end{aligned}$$

Therefore, if we take $N = 8 \times 10^5$, we should meet the accuracy requirement.

Marking:

- give them 3 marks showing that (21) is true,
- give them 2 mark for using some "reasonable" confidence interval to compute an approximation to N.
  If you took off marks in part (a) for not including p or $\hat{p}$ on the right side of (22)), don't take off marks again here for not including p or $\hat{p}$ properly in the computation of N.
  That is, just take off marks in part (a) for this error.

Of course, you can give them part marks for each of the points above if their answer is on the right track, but not completely correct.