1. The first 2 questions refer to the following experiment.

There are two identical bags. One contains 3 white and 1 black ball; the other 1 white and 3 black balls. You pick a bag randomly (probability $\frac{1}{2}$ for each bag) and then randomly pick one of the balls in the bag (probability $\frac{1}{4}$ for each ball). You got a white ball. Let X be the number of white balls in the *other* bag. *(The* **information** *that you got a white ball is very important.)*

What is $\mathbb{E}[X]$ (expected value)?

A 1

B $\frac{6}{4}$

C $\frac{10}{4}$

D 2

E $\frac{5}{4}$

2. What is $Var(X)$ (variance)?

A $\frac{2}{4}$

B $\frac{3}{4}$

C 1

D $\frac{5}{4}$

E $\frac{6}{4}$

3. A game costs \$$x$ to play. You toss 4 fair coins. If you get *more* heads than tails, you win and get back \$$10 + x$ for a *profit* of \$10. Otherwise, you lose and get nothing back, so your *loss* is \$$x$. What is an expression for your expected profit in dollars?

A $10 \times \frac{1}{2} - x \times \frac{1}{2}$

B $\frac{50 - 11x}{16}$

C $\frac{60 - 10x}{16}$

D $\frac{50 - x}{16}$

E $\frac{60 - x}{16}$

90 Minutes

4. A Martian couple continues to have children until they have 2 males (not necessarily in a row). On Mars, males are twice as likely as females. Assume children are *independent*. Let X be the number of children this couple will have. What is $\mathbb{E}[X]$, the expected number of children this couple will have?

A 2

B 3

C 2.5

D 3.5

E 4

5. You toss 5 independent fair coins. What is the probability that you will get 4 or more heads?

A $\binom{5}{4} \times \frac{1}{2^5}$

B $\frac{3}{16}$

C $\frac{5}{32}$

D $\frac{1}{4}$

E $\frac{9}{32}$

6. Step 1: Toss 9 fair coins. Step 2: if you got more heads than tails in Step 1, toss 9 more coins and stop; if you get fewer heads than tails in Step 1, stop. Let X be the number of heads you toss. What is $\mathbb{E}[X]$?

A 6.25

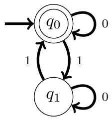
B 6.75

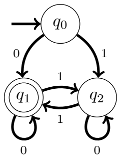
C 7.25

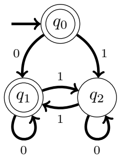
D 9

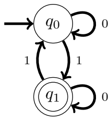
E 8

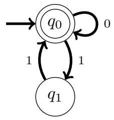
7. Language $\mathcal{L}_1$ = {all *non-empty* strings in which the number of 1's is even}. Which finite automaton solves this problem, i.e. the YES-set (set of accepted strings) for the automaton is $\mathcal{L}_1$?

A 

B 

C 

D 

E 

8. Language $\mathcal{L}_2$ = {*all* strings in which the number of 1's is even} which CFG solves this problem - i.e., generates the strings in $\mathcal{L}_2$?

A $S \to \varepsilon \mid 0S \mid S0 \mid 11S \mid S11$

B $S \to \varepsilon \mid 0S \mid S0 \mid 1S1$

C $S \to \varepsilon \mid 0S \mid 11S$

D $S \to \varepsilon \mid 1S \mid S1 \mid 0S0$

E $S \to \varepsilon \mid 0 \mid 11 \mid SS$

9. Which of the following is *countable*?

A The set of real numbers.

B A language (a possibly infinite set of *finite* strings).

C The set of all subsets of $\mathbb{N}$.

D The set of all functions from $\mathbb{R}$ to $\mathbb{R}$.

E The set of all functions from $\mathbb{N}$ to $\mathbb{N}$.

10. Which of the following is *not* a valid way to show that a set S is countable:

A Show an onto function from $\mathbb{N}$ to S.

B Show a 1-to-1 function from $\mathbb{N}$ to S.

C Show a bijection from $\mathbb{N}$ to S.

D Show there *does not exist* a 1-to-1 function from $\mathbb{N}$ to S.

E Show a 1-to-1 function from S to $\mathbb{N}$.

1. A random variable $\mathbf{X}$ has PDF shown on the right. Compute $\mathbb{E}[\mathbf{X}]$ (expectation) and $\sigma^2(\mathbf{X})$ (variance).

$\mathbf{X}$	-2	-1	0	1	2
$P_{\mathbf{X}}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{2}{10}$	$\frac{3}{10}$	$\frac{3}{10}$

A $\mathbb{E}[\mathbf{X}] = 0$ $\quad \sigma^2(\mathbf{X}) = 2$

B $\mathbb{E}[\mathbf{X}] = 0.6$ $\quad \sigma^2(\mathbf{X}) = 1$

C $\mathbb{E}[\mathbf{X}] = 0.6$ $\quad \sigma^2(\mathbf{X}) = 1.64$

D $\mathbb{E}[\mathbf{X}] = 1$ $\quad \sigma^2(\mathbf{X}) = 1.8$

E $\mathbb{E}[\mathbf{X}] = 0.5$ $\quad \sigma^2(\mathbf{X}) = 1.65$

2. $\mathbf{X}$ and $\mathbf{Y}$ are rolls of two independent fair dice and $\mathbf{Z} = \mathbf{X} + 2\mathbf{Y}$. Compute $\mathbb{E}[\mathbf{Z}]$ (expectation).

A 6

B 7

C $9\frac{1}{2}$

D $10\frac{1}{2}$

E 14

3. $\mathbf{X}$ and $\mathbf{Y}$ are rolls of two independent fair dice and $\mathbf{Z} = \mathbf{X} + 2\mathbf{Y}$. Compute $\sigma^2(\mathbf{Z})$ (variance). (The variance of a *single* die roll is 35/12.)

A 70/12

B 105/12

C 140/12

D 175/12

E 210/12

4. For a random variable $\mathbf{X}$, what does the standard deviation $\sigma(\mathbf{X})$ measure?

A The average value of $\mathbf{X}$ you will observe if you ran the experiment many times.

B The number of times you run the experiment (on average) before you observe the value $\mathbb{E}[\mathbf{X}]$.

C The size of the deviation between the observed value of $\mathbf{X}$ and the expected value $\mathbb{E}[\mathbf{X}]$.

D The probability that $\mathbf{X}$ will be larger than its expected value $\mathbb{E}[\mathbf{X}]$.

E The number of possible values of $\mathbf{X}$.

5. You toss two coins. If you get HH for the tosses, you roll 12 dice and *count* the number of sixes as $\mathbf{X}$. If you do not get HH for the tosses, you roll 6 dice and *count* the number sixes as $\mathbf{X}$. Compute $\mathbb{E}[\mathbf{X}]$, the expected number of sixes rolled.

A 1.25

B 1.5

C 1.75

D 2

E 2.5

6. **[Hard]** A Martian couple continues to have children until they have 2 males *in a row*. On Mars, *males are half as likely as females*. Assume children are *independent*. Let $\mathbf{X}$ be the number of children this couple will have. Compute $\mathbb{E}[\mathbf{X}]$, the expected number of children this couple will have.

A 9

B 10

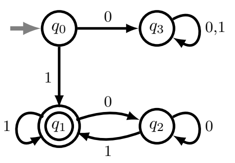
C 11

D 12

E 13

60 Minutes

7. The DFA on the right solves a computing problem defined by its YES-set (the language it accepts). What is a regular expression for this computing problem?



A $\{0,1\}^*$

B $\{1\}\bullet\{0,1\}^*\bullet\{1\}$

C $\{1\}\bullet\{\{0,1\}^*\bullet 1\}^*$

D $\{1\}^*$

E $\{1\}\bullet\{01\}^*$

8. What is the computing problem $\mathcal{L}$ solved by the CFG on the right? (What is the language generated?)

$S \to \varepsilon \mid 0 \mid 1 \mid SS$

A $\mathcal{L} = \{$all finite binary strings$\} = \Sigma^*$.

B $\mathcal{L} = \{$all finite binary strings of even length$\}$.

C $\mathcal{L} = \{$all finite binary strings of odd length$\}$.

D $\mathcal{L} = \{$all finite binary strings with an equal number of 0's and 1's$\}$.

E $\mathcal{L} = \{$all finite binary strings which contain 01 as a substring$\}$.

9. Which of the following is *countable*:
(I) Integers, $\mathbb{Z}$ (II) Valid C^{++} programs (III) The prime numbers

A I, II, III.

B only I, II.

C only I, III.

D only II, III.

E only I.

10. Rank deterministic finite automata (DFA), context free grammars (CFG), which are related to pushdown automata, and Turing Machines (TM) in order of how powerful they are. (For example, $DFA > CFG$ if DFAs can solve more problems that CFGs; $DFA = CFG$ if DFAs and CFGs can solve the same problems; $DFA < CFG$ if DFAs can solve fewer problems that CFGs.

A $DFA > CFG > TM$

B $DFA = CFG > TM$

C $DFA = CFG = TM$

D $DFA = CFG < TM$

E $DFA < CFG < TM$

SCRATCH

1. Which of the following describes the expected value of a random variable $\mathbf{X}$?

$\boxed{\text{A}}$ It is the typical observed value of $\mathbf{X}$ in an experiment.

$\boxed{\text{B}}$ It is the most likely observed value of $\mathbf{X}$ in an experiment.

$\boxed{\text{C}}$ It is one of the possible observed values of $\mathbf{X}$ in an experiment.

$\boxed{\text{D}}$ It is the maximum value of $\mathbf{X}$ that can be observed in an experiment.

$\boxed{\text{E}}$ None of the above.

2. For a random variable $\mathbf{X}$, what does the standard deviation $\sigma(\mathbf{X})$ measure?

$\boxed{\text{A}}$ The average value of $\mathbf{X}$ you will observe if you ran the experiment many times.

$\boxed{\text{B}}$ The number of times you run the experiment (on average) before you observe the value $\mathbb{E}[\mathbf{X}]$.

$\boxed{\text{C}}$ The size of the deviation between the observed value of $\mathbf{X}$ and the expected value $\mathbb{E}[\mathbf{X}]$.

$\boxed{\text{D}}$ The probability that $\mathbf{X}$ will be larger than its expected value $\mathbb{E}[\mathbf{X}]$.

$\boxed{\text{E}}$ The number of possible values of $\mathbf{X}$.

3. A real valued $\mathbf{X}$ has expectation $\mathbb{E}[\mathbf{X}] = \mu$. Which is *not* a valid formula for the variance $\sigma^2(\mathbf{X})$?

$\boxed{\text{A}}$ $\mathbb{E}[(\mathbf{X} - \mu)^2]$.

$\boxed{\text{B}}$ $\mathbb{E}[\mathbf{X}^2] - 2\mu\,\mathbb{E}\,[\mathbf{X}] + \mathbb{E}[\mathbf{X}]^2$.

$\boxed{\text{C}}$ $\mathbb{E}[\mathbf{X}^2] - \mu^2$.

$\boxed{\text{D}}$ $\mathbb{E}[\,|\mathbf{X}|^2\,] - \mu^2$.

$\boxed{\text{E}}$ They are all valid.

4. A class has 10 students. Each student is given a random number in $\{1, 2, 3, \ldots, 10\}$. The score $\mathbf{X}$ for the class is now computed as follows. For every pair of students whose numbers match, the number is added *once* to the score. For example, if the numbers given to the students are $\{1, 1, 1, 2, 3, 4, 5, 8, 10, 10\}$, then the score $\mathbf{X} = 13$. What is an approximate value for $\mathbb{E}[\mathbf{X}]$? *[Hint: Linearity of expected value.]*

$\boxed{\text{A}}$ $\mathbb{E}[\mathbf{X}] \approx 10$.

$\boxed{\text{B}}$ $\mathbb{E}[\mathbf{X}] \approx 25$.

$\boxed{\text{C}}$ $\mathbb{E}[\mathbf{X}] \approx 55$.

$\boxed{\text{D}}$ $\mathbb{E}[\mathbf{X}] \approx 105$.

$\boxed{\text{E}}$ $\mathbb{E}[\mathbf{X}] \approx 155$.

5. A random variable $\mathbf{X}$ has PDF shown on the right. Compute $\mathbb{E}[\mathbf{X}]$ (expectation) and $\sigma^2(\mathbf{X})$ (variance).

$\mathbf{X}$	-2	-1	0	1	2
$P_{\mathbf{X}}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$

- A $\mathbb{E}[\mathbf{X}] = 0$ $\quad\sigma^2(\mathbf{X}) = 2$
- B $\mathbb{E}[\mathbf{X}] = 0$ $\quad\sigma^2(\mathbf{X}) = 4$
- C $\mathbb{E}[\mathbf{X}] = 1$ $\quad\sigma^2(\mathbf{X}) = 4$
- D $\mathbb{E}[\mathbf{X}] = 1$ $\quad\sigma^2(\mathbf{X}) = 8$
- E None of the above.

6. For the random variable $\mathbf{X}$ in Problem 5 above, let $\mathbf{Y} = 2\mathbf{X} + 1$. Compute $\mathbb{E}[\mathbf{Y}]$ and $\sigma^2(\mathbf{Y})$.

- A $\mathbb{E}[\mathbf{X}] = 0$ $\quad\sigma^2(\mathbf{X}) = 2$
- B $\mathbb{E}[\mathbf{X}] = 0$ $\quad\sigma^2(\mathbf{X}) = 4$
- C $\mathbb{E}[\mathbf{X}] = 1$ $\quad\sigma^2(\mathbf{X}) = 4$
- D $\mathbb{E}[\mathbf{X}] = 1$ $\quad\sigma^2(\mathbf{X}) = 8$
- E None of the above.

7. **[Hard]** A Martian couple continues to have children until they have 2 males *in a row*. On Mars, *males are twice as likely as females*. Assume children are *independent*. Let $\mathbf{X}$ be the number of children this couple will have. Compute $\mathbb{E}[\mathbf{X}]$, the expected number of children this couple will have.

- A $2\frac{1}{4}$.
- B $3\frac{3}{4}$.
- C 6.
- D 12.
- E None of the above.

8. Which (if any) of the following sets *do not* have the same cardinality as $\mathbb{N} = \{1, 2, 3, 4, 5, \ldots\}$?

- A $\{0, 1, 2, 3, 4, 5\}$.
- B The rationals, $\mathbb{Q} = \{\frac{z}{n} \mid z \in \mathbb{Z}, n \in \mathbb{N}\}$.
- C The set of valid C^{++} programs.
- D The set of all possible Turing Machines.
- E They all have the same cardinality as $\mathbb{N}$.

9. Which (if any) of the following sets is **not** *countable*?

- A $\{0, 1, 2, 3, 4, 5\}$.
- B The rationals, $\mathbb{Q} = \{\frac{z}{n} \mid z \in \mathbb{Z}, n \in \mathbb{N}\}$.
- C The set of valid C^{++} programs.
- D The set of all possible Turing Machines.
- E They are all countable.

10. Which (if any) is *not* a valid way to prove that a set S is countable?

- A Show an injection exists from S to $\mathbb{N}$.
- B Show a 1-to-1 function exists from S to $\mathbb{N}$.
- C Show a surjection exists from $\mathbb{N}$ to S.
- D Show that S is finite.
- E They are all valid ways to show S is countable.

11. Which of the following strings is *not* in the language described by the regular expression $\{0, 10\}^*$?

- A ε.
- B 010010.
- C 100100.
- D 010110.
- E They are all in the language.

12. Which computing problem (if any) *cannot* be solved by a DFA (deterministic finite automata)?

A $\mathcal{L} = \{\text{strings with at least one } 1\}$.

B $\mathcal{L} = \{(01)^{\bullet n} \mid n \geq 0\}$.

C $\mathcal{L} = \{\text{strings that end with } 01\}$.

D $\mathcal{L} = \{\text{strings with } \textit{more} \text{ 1s than 0s}\}$.

E They can each be solved by some DFA.

13. Which problem (if any) *cannot* be solved by a CFG (context free grammar)?

A $\mathcal{L} = \{\text{strings with at least one } 1\}$.

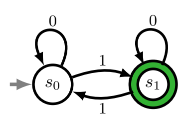
B $\mathcal{L} = \{(01)^{\bullet n} \mid n \geq 0\}$.

C $\mathcal{L} = \{\text{strings that end with } 01\}$.

D $\mathcal{L} = \{\text{strings with } \textit{more} \text{ 1s than 0s}\}$.

E They can each be solved by some CFG.

14. The DFA on the right solves a computing problem defined by its (YES)-set (the language it accepts). The accept state is s_1. What is a regular expression for this computing problem?



A $\{0,1\}^*$.

B $\{0,1\}^* \bullet 1$.

C $\{0\}^* \bullet 1 \bullet \{\{0\}^* \bullet 1 \bullet \{0\}^* \bullet 10\}^*$

D $\{0\}^* \bullet 1 \bullet \{\{0\}^* \bullet 1 \bullet \{0\}^* \bullet 1\}^* \bullet \{0\}^*$

E None of the above.

15. Rank deterministic finite automata (DFA), context free grammars (CFG), which are related to pushdown automata, and Turing Machines (TM) in order of how powerful they are. (For example, $DFA > CFG$ if DFAs can solve more problems that CFGs; $DFA = CFG$ if DFAs and CFGs can solve the same problems; $DFA < CFG$ if DFAs can solve fewer problems that CFGs.

A $DFA > CFG > TM$

B $DFA = CFG > TM$

C $DFA = CFG = TM$

D $DFA = CFG < TM$

E $DFA < CFG < TM$

1. A function f maps $\{a,b,c,d\}$ to $\{1,2,3\}$ as follows: $f(a) = 1, f(b) = 2, f(c) = 1, f(d) = 2$.

A f is injective (1-to-1) but not bijective.

B f is surjective (onto) but not bijective.

C f is bijective (1-to-1 and onto).

D f is neither injective nor surjective.

E f is not a valid function.

2. A set $\mathcal{S}$ contains all the distinct functions which map $\mathbb{N}$ to $\{0\}$. What is the cardinality (size) of $\mathcal{S}$?

A 0.

B 1.

C Bigger than 1 but finite.

D The same as $|\,\mathbb{N}\,|$.

E Strictly larger than $|\,\mathbb{N}\,|$.

3. A set $\mathcal{S}$ contains all the distinct functions which map $\mathbb{N}$ to $\{2,3,4\}$. What is the cardinality (size) of $\mathcal{S}$?

A 0.

B 1.

C Bigger than 1 but finite.

D The same as $|\,\mathbb{N}\,|$.

E Larger than $|\,\mathbb{N}\,|$.

4. What is the cardinality (size) of the set containing all distinct python programs of finite length?

A 0.

B 1.

C Bigger than 1 but finite.

D The same as $|\,\mathbb{N}\,|$.

E Larger than $|\,\mathbb{N}\,|$.

5. Which set is *not* countable, i.e., has a cardinality strictly larger than $|\,\mathbb{N}\,|$?

A $\mathbb{Q}$, the rational numbers.

B All distinct finite binary strings.

C The set of all possible Turing Machines.

D The set containing all distinct functions that map $\{0,1\}$ to $\mathbb{N}$.

E They are all countable.

6. What is a computing problem?

A A person who knows how to write a program in python.

B A machine that transitions between states.

C A rule for deciding if a string belongs to a set.

D Any set of finite binary strings.

E A Turing Machine.

7. $\mathcal{L}$ is a computing problem. What can we say about the cardinality (size) of $\mathcal{L}$?

A $\mathcal{L}$ must have finite cardinality.

B $\mathcal{L}$ must have infinite cardinality.

C $\mathcal{L}$ must be countable.

D $\mathcal{L}$ must be uncountable.

E None of the above.

8. $\mathcal{L}_1$ and $\mathcal{L}_2$ are computing problems. Which of the following is *not* a computing problem?

A $\mathcal{L}_1 \bullet \mathcal{L}_2$.

B $\mathcal{L}_1^*$.

C $\mathcal{L}_1 \cap \mathcal{L}_2^*$.

D $\mathcal{L}_1^* \cup \overline{\mathcal{L}_2^*}$.

E They are all computing problems

9. Which of the following strings is in the language described by the regular expression $\{0, 11\}^*$?

A 011111.

B 010010.

C 100100.

D 110011.

E None of the strings above are in the language.

10. Which computing problem *cannot* be solved by a DFA (deterministic finite automata)?

A $\mathcal{L} = \{\text{strings with at least one 1}\}$.

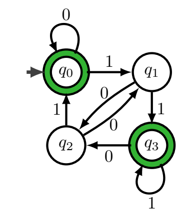
B $\mathcal{L} = \{(01)^{\bullet n} \mid n \geq 0\}$.

C $\mathcal{L} = \{\text{strings that end with 101}\}$.

D $\mathcal{L} = \{1^{\bullet n} w | n \geq 1 \text{ and } w \text{ has } n \text{ or more 1s}\}$.

E Each problem can be solved by a DFA.

11. For the DFA on the right, what sequence of states are visited for the input 0110.



A $q_2\ q_0\ q_1\ q_3\ q_2$.

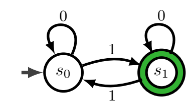
B $q_0\ q_0\ q_0\ q_0\ q_0$.

C $q_0\ q_1\ q_3\ q_2$.

D $q_0\ q_0\ q_1\ q_2\ q_3$.

E $q_0\ q_0\ q_1\ q_3\ q_2$.

12. How many 4-bit strings are in the YES-set of the DFA on the right. Accept states are double circles (only s_1 in this case).



A 6.

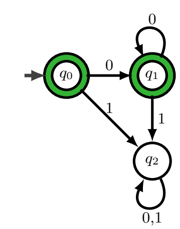
B 7.

C 8.

D 9.

E 10.

13. Which is *not* a description of the computing problem solved by the DFA on the right? Accept states are double circles (q_0, q_1 in this case).



A $\{0\}^*$.

B $\{0^{\bullet n} | n \geq 0\}$.

C All strings with no ones, including the empty string ε.

D Strings generated by the CFG: $S \to \varepsilon | 0S$.

E They all describe the computing problem solved by the DFA.

14. Which string *cannot* be generated by the CFG: $S \to 0|1|0S0|1S1$.

A 01110

B 10001

C 0110

D 11111

E They can all be generated.

15. Describe the strings generated by the CFG on the right.

$$1 : S \to A|B$$
$$2 : A \to 0|0A$$
$$3 : B \to 1|1A$$

A Nonempty strings containing only 0s.

B Nonempty strings containing only 1s.

C All nonempty strings.

D Nonempty strings which contain either only 0s or only 1s.

E None of the above

16. Which is a true statement about the computing problem $\mathcal{L} = \{w\#w | w \in \{0,1\}^*\}$. (# is punctuation)

- A A DFA can solve $\mathcal{L}$. A Turing Machine can solve $\mathcal{L}$.
- B A DFA cannot solve $\mathcal{L}$. A Turing Machine cannot solve $\mathcal{L}$.
- C A DFA can solve $\mathcal{L}$. A Turing Machine cannot solve $\mathcal{L}$.
- D A DFA cannot solve $\mathcal{L}$. A Turing Machine can solve $\mathcal{L}$.
- E None of the above.

17. How do we know there are computing problems which Turing Machines cannot solve?

- A Because the Turing Machines are countable and the computing problems are countable.
- B Because the Turing Machines are uncountable and the computing problems are countable.
- C Because the Turing Machines are countable and the computing problems are uncountable.
- D Because the Turing Machines are uncountable and the computing problems are uncountable.
- E None of the above proves there are computing problems which Turing Machines cannot solve.

18. What is the difference between a decider D for a language $\mathcal{L}$ and a recognizer R for $\mathcal{L}$.

- A For $w \in \mathcal{L}$, D halts with YES but R may infinite loop.
- B For $w \in \mathcal{L}$, D halts with YES and R halts but may say NO.
- C For $w \notin \mathcal{L}$, D halts with NO but R *may* sometimes go into an infinite loop.
- D For $w \notin \mathcal{L}$, D halts with NO but R *must* always go into an infinite loop.
- E For $w \notin \mathcal{L}$, D halts with NO and R halts but may say YES.

19. What is the Ultimate Debugger which we discussed in class?

- A A program that solves Goldbach's conjecture.
- B A program that solves the twin-prime conjecture.
- C A program that determines if another program will compile under a C^{++} compiler.
- D A program that translates another program into binary machine-code.
- E A program that determines if another program when run will halt.

20. Which answer is a valid conclusion about the decidability of the language $\mathcal{L}_B$?

- A $\mathcal{L}_A$ is decidable. A decider for $\mathcal{L}_B$ can be converted to a decider for $\mathcal{L}_A$. So, $\mathcal{L}_B$ is decidable.
- B $\mathcal{L}_A$ is decidable. A decider for $\mathcal{L}_A$ can be converted to a decider for $\mathcal{L}_B$. So, $\mathcal{L}_B$ is undecidable.
- C $\mathcal{L}_A$ is undecidable. A decider for $\mathcal{L}_A$ can be converted to a decider for $\mathcal{L}_B$. So, $\mathcal{L}_B$ is undecidable.
- D $\mathcal{L}_A$ is undecidable. A decider for $\mathcal{L}_B$ can be converted to a decider for $\mathcal{L}_A$. So, $\mathcal{L}_B$ is decidable.
- E $\mathcal{L}_A$ is undecidable. A decider for $\mathcal{L}_B$ can be converted to a decider for $\mathcal{L}_A$. So, $\mathcal{L}_B$ is undecidable.

1. How many injective (1-to-1) functions map $\{a, b, c, d\}$ to $\{1, 2, 3\}$?

- A 0.
- B 36.
- C 42.
- D 81.
- E None of the above.

2. How many surjective (onto) functions map $\{a, b, c, d\}$ to $\{1, 2, 3\}$?

- A 0.
- B 36.
- C 42.
- D 81.
- E None of the above.

3. An injective function f maps a set $\mathcal{A}$ to $\mathbb{N}$, $f : \mathcal{A} \mapsto \mathbb{N}$. Which is not true?

- A $\mathcal{A}$ can be finite.
- B $\mathcal{A}$ can be infinite.
- C $\mathcal{A}$ must be a subset of $\mathbb{N}$, $\mathcal{A} \subseteq \mathbb{N}$.
- D $\mathcal{A}$ can be the set of all possible finite computer programs in python.
- E All of the above is true.

4. A computing problem is a language. The cardinality of the set of all possible computing problems is:

- A Finite.
- B Countable.
- C Infinite but countable.
- D Uncountable.
- E None of the above.

5. The language $\mathcal{L} = \{0, 00, 000\} \bullet \{\varepsilon, 1, 11\}$. Which string is not in $\mathcal{L}$?

- A 0.
- B 011.
- C 100.
- D 001.
- E They are all in $\mathcal{L}$.

6. For languages $\mathcal{L}_1 = \{1\}^*$ and $\mathcal{L}_2 = \{1\}\bullet\{0,1\}^*$, which is true? ($\{\}^*$ is Kleene star.)

$\boxed{\text{A}}$ $\mathcal{L}_1 \subseteq \mathcal{L}_2$.

$\boxed{\text{B}}$ $\mathcal{L}_2 \subseteq \mathcal{L}_1$.

$\boxed{\text{C}}$ $\mathcal{L}_1 = \mathcal{L}_2$.

$\boxed{\text{D}}$ The regular expressions describing $\mathcal{L}_1$ and $\mathcal{L}_2$ are not valid regular expressions.

$\boxed{\text{E}}$ None of the above are true.

7. Which regular expression describes all the strings with at least two bits? ($\Sigma = \{0,1\}$.)

$\boxed{\text{A}}$ $\Sigma\bullet\Sigma$.

$\boxed{\text{B}}$ Σ^*.

$\boxed{\text{C}}$ $\Sigma^*\bullet\Sigma^*$.

$\boxed{\text{D}}$ $(\Sigma\bullet\Sigma)^*$.

$\boxed{\text{E}}$ None of the above.

8. What is the final resting state for the DFA with input 110010.

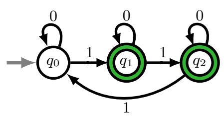
$\boxed{\text{A}}$ q_0.

$\boxed{\text{B}}$ q_1.

$\boxed{\text{C}}$ q_2.

$\boxed{\text{D}}$ This is not a valid DFA.

$\boxed{\text{E}}$ None of the above.



9. How many 6 bit strings are in the YES-set of the DFA in problem 8.

$\boxed{\text{A}}$ 19.

$\boxed{\text{B}}$ 22.

$\boxed{\text{C}}$ 39.

$\boxed{\text{D}}$ 42.

$\boxed{\text{E}}$ None of the above.

10. Which is the computing problem solved by the DFA in problem 8

$\boxed{\text{A}}$ $\mathcal{L} = \{$strings with a number of 1s divisible by 3$\}$.

$\boxed{\text{B}}$ $\mathcal{L} = \{$strings with a number of 1s not divisible by 3$\}$.

$\boxed{\text{C}}$ $\mathcal{L} = \{$strings with three more 1s than 0s.$\}$.

$\boxed{\text{D}}$ $\mathcal{L} = \{$strings with three more 0s than 1s$\}$.

$\boxed{\text{E}}$ None of the above.

11. Which computing problem *cannot* be solved by a DFA (deterministic finite automata)?

$\boxed{\text{A}}$ $\mathcal{L} = \{$strings with no 1s$\}$.

$\boxed{\text{B}}$ $\mathcal{L} = \{$strings with no 1s or an even number of 0s$\}$.

$\boxed{\text{C}}$ $\mathcal{L} = \{$strings with a number of 1s *not* divisible by 3$\}$.

$\boxed{\text{D}}$ $\mathcal{L} = \{$strings which begin and end in different bits$\}$.

$\boxed{\text{E}}$ Each problem above can be solved by a DFA.

12. The main limitation of the DFA which prevents it from solving $\mathcal{L} = \{0^n1^n | n \geq 0\}$ is:

$\boxed{\text{A}}$ The DFA is not a very fast machine so it would take too long.

$\boxed{\text{B}}$ The DFA can't have more than one yes-state.

$\boxed{\text{C}}$ The input string can be arbitrarily long.

$\boxed{\text{D}}$ The DFA can go into an infinite loop.

$\boxed{\text{E}}$ The DFA cannot remember how many 0s have gone by because it has only finitely many states.

13. Which string *cannot* be generated by the CFG: $S \to \varepsilon \mid 0 \mid 0S$.

$\boxed{\text{A}}$ ε.

$\boxed{\text{B}}$ 00.

$\boxed{\text{C}}$ 000.

$\boxed{\text{D}}$ 0001.

$\boxed{\text{E}}$ They can all be generated.

14. Which string cannot be generated by the CGF shown?

$1: S \to B1A \mid B1A1B$
$2: A \to \varepsilon \mid B1B1B1B \mid AA$
$3: B \to \varepsilon \mid 0B$

$\boxed{\text{A}}$ 011101

$\boxed{\text{B}}$ 110101

$\boxed{\text{C}}$ 111100

$\boxed{\text{D}}$ 011100

$\boxed{\text{E}}$ They can all be generated.

15. What is the difference between a Turing machine decider and a Turing machine recognizer?

$\boxed{\text{A}}$ Both are the same thing.

$\boxed{\text{B}}$ A decider cannot write to the tape, a recognizer can.

$\boxed{\text{C}}$ A decider can write to the tape, a recognizer cannot.

$\boxed{\text{D}}$ A decider has a finite number of states, a recognizer can have infinitely many states.

$\boxed{\text{E}}$ A decider must always halt, saying YES or NO. A recognizer may not halt.

16. Consider the computing problem $\mathcal{L} = \{w\#w | w \in \{0,1\}^*\}$ (# is punctuation). Which claim is true?

[A] A DFA can solve $\mathcal{L}$.

[B] A DFA with a top-access stack can solve $\mathcal{L}$.

[C] A Turing machine decider can solve $\mathcal{L}$.

[D] A Turing machine decider cannot solve $\mathcal{L}$.

[E] None of the above.

17. The theory of computing and the Church-Turing thesis define computing problems and algorithms as:

[A] A computing problem is a string. An algorithm is a recognizer.

[B] A computing problem is a set of finite binary strings. An algorithm is a recognizer.

[C] A computing problem is a Turing Machine. An algorithm is a decider.

[D] A computing problem is a set of finite binary strings. An algorithm is a person.

[E] A computing problem is a set of finite binary strings. An algorithm is a decider.

18. The Ultimate Debugger, which we discussed in class solves, what problem?

[A] $\mathcal{L} = \{\langle M \rangle \# w | M$ halts on input $w\}$.

[B] $\mathcal{L} = \{\langle M \rangle \# w | M$ does not halt on input $w\}$.

[C] $\mathcal{L} = \{\langle M \rangle | M$ halts and says yes on some input$\}$.

[D] $\mathcal{L} = \{\langle M \rangle | M$ halts and says no on some input$\}$.

[E] None of the above.

19. Any decider for problem $\mathcal{L}_A$ can be used to decide problem $\mathcal{L}_B$. Which conclusion is not true?

[A] We found out $\mathcal{L}_A$ is decidable. We concluded $\mathcal{L}_B$ must be decidable.

[B] We found out $\mathcal{L}_A$ is undecidable. We concluded $\mathcal{L}_B$ could still be decidable.

[C] We found out $\mathcal{L}_B$ is decidable. We concluded $\mathcal{L}_A$ could still be undecidable.

[D] We found out $\mathcal{L}_B$ is undecidable. We concluded $\mathcal{L}_A$ must be undecidable.

[E] All of the above are true.

20. Let $\mathcal{M}$ be the set of all possible Turing Machines. Which statement is not true?

[A] Every Turing Machine in $\mathcal{M}$ can be uniquely encoded into a finite binary string.

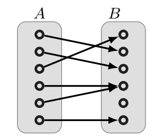
[B] All Turing Machines in $\mathcal{M}$ can be listed: $\{\langle M_1 \rangle, \langle M_2 \rangle, \langle M_3 \rangle, \langle M_4 \rangle, \ldots, \}$.

[C] $\mathcal{M}$ is countable.

[D] There is an injection from $\mathcal{M}$ to $\mathbb{N}$.

[E] All of the above are true.

1. Which describes the function on the right that maps A to B.



[A] f is not an injection (1-to-1) and f is not a surjection (onto).

[B] f is an injection (1-to-1) and f is not a surjection (onto).

[C] f is not an injection (1-to-1) and f is a surjection (onto).

[D] f is an injection (1-to-1) and f is a surjection (onto).

[E] None of the above.

2. A set $\mathcal{S}$ contains all the distinct functions which map $\{0,1\}$ to $\mathbb{N}$. What is the cardinality of $\mathcal{S}$?

[A] 0.

[B] 1.

[C] Bigger than 1 but finite.

[D] The same as $|\mathbb{N}|$.

[E] Strictly larger than $|\mathbb{N}|$.

3. A set $\mathcal{S}$ contains all the distinct functions which map $\mathbb{N}$ to $\{0,1\}$. What is the cardinality of $\mathcal{S}$?

[A] 0.

[B] 1.

[C] Bigger than 1 but finite.

[D] The same as $|\mathbb{N}|$.

[E] Strictly larger than $|\mathbb{N}|$.

4. What is a computing problem?

[A] A person who knows how to write a program in python.

[B] A machine that transitions between states.

[C] A rule for deciding if a string belongs to a set.

[D] Any set of finite binary strings.

[E] A Turing Machine.

5. Which set is *not* countable, i.e., has a cardinality strictly larger than $|\mathbb{N}|$?

[A] $\mathbb{Q}$, the rational numbers.

[B] All distinct finite binary strings.

[C] All possible Turing Machines.

[D] All possible computing problems.

[E] They are all countable.

6. The language $\mathcal{L} = \{11, 111\}^*$ (Kleene star). Which string is not in $\mathcal{L}$?

A ε.

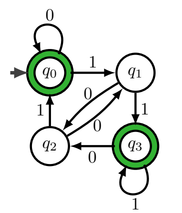
B 1.

C 1111.

D 11111.

E They are all in $\mathcal{L}$.

7. What is the final resting state for the DFA on input 110110.



A q_0.

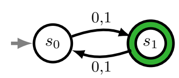
B q_1.

C q_2.

D q_3.

E None of the above

8. Let $\Sigma = \{0, 1\}$. Which regular expression is the problem solved by the DFA.



A Σ^*.

B $\Sigma \bullet \Sigma \bullet \Sigma$.

C $(\Sigma \bullet \Sigma) \bullet \Sigma^*$.

D $\Sigma \bullet (\Sigma \bullet \Sigma)^*$.

E None of the above.

9. How many 5 bit strings are in the (YES)-set of the DFA in problem 8.

A 4.

B 8.

C 16.

D 32.

E None of the above.

10. Which computing problem *cannot* be solved by a DFA (deterministic finite automata)?

A $\mathcal{L} = \{$strings with no 1s$\}$.

B $\mathcal{L} = \{$strings with an odd number of 1s$\}$.

C $\mathcal{L} = \{$strings that are *not* 1111$\}$.

D $\mathcal{L} = \{$strings with more 1s than 0s$\}$.

E Each problem above can be solved by a DFA.

11. The main limitation of the DFA which prevents it from solving $\mathcal{L} = \{0^{\bullet n}1^{\bullet n+3} | n \geq 0\}$ is:

A The DFA can't have more than one yes-state.

B The input string can be arbitrarily long.

C The DFA can go into an infinite loop.

D The DFA cannot remember how many 0s have gone by because it has only finitely many states.

E None of the above, because a DFA can solve $\mathcal{L}$.

12. Which string cannot be generated by the CGF shown?

$1: S \to \varepsilon \mid A \mid B$
$2: A \to 0 \mid 0B$
$3: B \to 1 \mid 1A$

A ε

B 010

C 101

D 011

E They can all be generated.

13. Which CFG generates all strings with an even number of bits, including ε.

A $S \to \varepsilon \mid SS$

B $S \to \varepsilon \mid 0 \mid 1 \mid SS$

C $S \to \varepsilon \mid 01S$

D $S \to \varepsilon \mid 00S \mid 01S \mid 10S \mid 11S$

E None of the above.

14. Which comparison between DFAs and CFGs is correct?

A A DFA can solve language $\mathcal{L}$ if and only if a CFG can generate language $\mathcal{L}$.

B If a DFA can solve language $\mathcal{L}$, then a CFG can generate language $\mathcal{L}$.

C If a CFG can generate language $\mathcal{L}$, then a DFA can solve language $\mathcal{L}$.

D There is some language $\mathcal{L}$ which a DFA can solve, but no CFG can generate that language $\mathcal{L}$.

E None of the above.

15. In the theory of computing, we define computing problems and algorithms as:

A A computing problem is a string. An algorithm is a recognizer.

B A computing problem is a set of finite binary strings. An algorithm is a recognizer.

C A computing problem is a Turing Machine. An algorithm is a decider.

D A computing problem is a set of finite binary strings. An algorithm is a person.

E A computing problem is a set of finite binary strings. An algorithm is a decider.

16. Why do we prefer a Turing machine decider over a Turing machine recognizer?

$\boxed{\text{A}}$ Because there are some yes sets that are accepted by a decider but not a recognizer.

$\boxed{\text{B}}$ Because a decider can write to the tape, but a recognizer cannot.

$\boxed{\text{C}}$ Because a decider has a finite number of states, but a recognizer has infinitely many states.

$\boxed{\text{D}}$ Because any useful algorithm should always halt giving an answer.

$\boxed{\text{E}}$ We don't prefer one over the other because both are the same thing.

17. Consider the computing problem $\mathcal{L} = \{0^{\bullet m}1^{\bullet n}0^{\bullet k} \mid m, n, \geq 0 \text{ and } n = m + k\}$. Which claim is not true?

$\boxed{\text{A}}$ A DFA cannot solve $\mathcal{L}$.

$\boxed{\text{B}}$ A DFA with an external top-access stack memory can solve $\mathcal{L}$.

$\boxed{\text{C}}$ A CFG can generate $\mathcal{L}$.

$\boxed{\text{D}}$ A Turing machine decider can solve $\mathcal{L}$.

$\boxed{\text{E}}$ None of the above.

18. Which problem is not solvable by an algorithm?

$\boxed{\text{A}}$ $\mathcal{L} = \{\langle M \rangle \mid M \text{ is a valid Turing Machine.}\}$

$\boxed{\text{B}}$ $\mathcal{L} = \{0^{\bullet n} \mid n \geq 0\}$.

$\boxed{\text{C}}$ $\mathcal{L} = \{0^{\bullet 2^n} \mid n \geq 0\}$.

$\boxed{\text{D}}$ Determining if any given python program correctly says if an input n is prime or not.

$\boxed{\text{E}}$ None of the above.

19. Problem $\mathcal{L}_A$ is reducible to $\mathcal{L}_B$, that is $\mathcal{L}_A \leq_{\text{R}} \mathcal{L}_B$. We know that $\mathcal{L}_B$ is decidable. Which is true?

$\boxed{\text{A}}$ $\mathcal{L}_A$ must be undecidable.

$\boxed{\text{B}}$ $\mathcal{L}_A$ can be undecidable.

$\boxed{\text{C}}$ $\mathcal{L}_A$ must be decidable.

$\boxed{\text{D}}$ $\mathcal{L}_A$ must be finite.

$\boxed{\text{E}}$ None of the above.

20. Let $\mathcal{M}$ be the set of all possible Turing Machines. Which statement is not true?

$\boxed{\text{A}}$ Every Turing Machine in $\mathcal{M}$ can be uniquely encoded into a finite binary string.

$\boxed{\text{B}}$ All Turing Machines in $\mathcal{M}$ can be listed: $\{\langle M_1 \rangle, \langle M_2 \rangle, \langle M_3 \rangle, \langle M_4 \rangle, \ldots, \}$.

$\boxed{\text{C}}$ $\mathcal{M}$ is countable.

$\boxed{\text{D}}$ Given any computing problem $\mathcal{L}$, there is a Turing Machine in $\mathcal{M}$ which solves $\mathcal{L}$.

$\boxed{\text{E}}$ All of the above are true.