PHYS 360

PS2. 09.05

0. Done

1. i) fair coins: $2^4 = 16$
   ii) five coins: $2^5 = 32$

2. i) four die : 6^4
   ii) five die : 6^5

3. $2^m \times 6^n \geq 26.$ $\Rightarrow$ 5 coins, or 2 dice, or 3 coins & one die

4.

(a) $2^4 \times 1 + 2^3 \times 0 + 2^2 \times 1 + 2^1 \times 1 + 2^0 \times 1$

$= 16 + 0 + 4 + 2 + 1 = 23$

(b) $2^7 \times 1 + 2^6 \times 1 + 2^5 \times 0 + 2^4 \times 0 + 2^3 \times 1 + 2^2 \times 0 + 2^1 \times 1 + 2^0 \times 0$

$= 128 + 64 + 0 + 0 + 8 + 0 + 2 + 0$

$= 130 + 72 = 202$

(c) 101101_2

$= 2^5 \times 1 + 2^4 \times 0 + 2^3 \times 1 + 2^2 \times 1 + 2^1 \times 0 + 2^0 \times 1$

$= 32 + 0 + 8 + 4 + 0 + 1$

$= 45$

5.

(a) $101010_2 = 32 + 0 + 8 + 0 + 2 + 0$

(b) $256 + 128 + 64 + 32 + 0 + 8 + 4 + 2 + 1$

$= 111101111_2$

(c) 19_{10}

$= 16 + 0 + 0 + 2 + 1$

$= 10011_2$

6.

(a)

A	B	Result
T	T	F
T	F	T
F	T	T
F	F	T

(b) NAND

reversed result from AND gate.

AND

T → F
F → T
F → T
F → T

7.

(a)

A	B	Result
T	T	F
T	F	F
F	T	F
F	F	T

(b) NOR

OR: T T T F → NOR: F F F T

8.

AND

(a)

A	B	A OR B	C	NAND C	OUTPUT
T	T	T	T	F	F
T	T	T	F	T	T
T	F	T	T	F	F
T	F	T	F	T	T
F	T	T	T	F	F
F	T	T	F	T	T
F	F	F	T	F	F
F	F	F	F	T	F

9.

(a) Ornes, S. (2025, March 17). Quantum Speedup Found Huge Class of Hard Problems. Quantamagazine. https://www.quantamagazine.org/quantum-speedup-found-for-huge-class-of-hard-problems-20250317/

(b) Ever since a quantum computer and Shor's factoring algorithm were introduced, there has been fierce competition between quantum and classical algorithms. Quantum algorithms that seemingly obtained "quantum advantage" have often been dequantized by their classical counterparts. Yet, new algorithm called DQI (decoded quantum interferometry), by Google Quantum AI remains unbeaten. It optimizes certain class of problem, by exploiting QFT and decoding methods to search better solutions, which correspond to those with higher quantum amplitudes.

(c)

1) further information about what DQI is.
2) specific applications of DQI
3) The degree of QPU to run DQI - will it be useful in near term?
4) Is there any particular way/rule to classify problems?
The structure/map of all the classes that have something to do with quantum algorithms (There should be classes regarding computational theory, but the only things I know are P, NP, NPC, NP-hard)