1.

a. $1\frac{12!}{(6!)^2 2!} = 462$

b. ${}_{12}C_4 = \frac{12!}{(4!)^3 3!} = 5775$

c. $\sum_{i=0}^{5}\binom{12}{i} + \binom{12}{6} \div 2 = 2048$

2.

a. The reasoning is that there are 8 letters in the total set and you can choose 3 E for their placement first. Then you have a remaining of 5 letters from which N appears twice, so you choose the number of ways to select a set of k items from a set of n distinct items

b. Similarly, we start off with 8 letter than since we have 3 E and we don't care about order we divide by 3! And likewise with the 2 N.

c. $\frac{\binom{8}{3}\binom{8}{2}3!}{8} = 420$

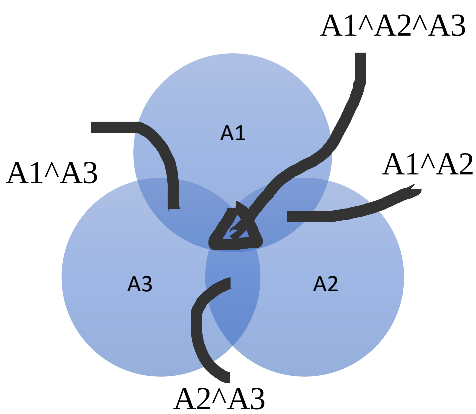
3.

a. ${}_{36}C_4 = 58905$

b. $\binom{16}{2}\binom{20}{2} = 22800$

4.

a.



$P(A_1) = 0.24$ $P(A_2) = 0.18$ $P(A_3) = 0.1$

$P(A_1 \cup A_2) = 0.3$ $P(A_1 \cup A_3) = 0.28$ $P(A_2 \cup A_3) = 0.24$

$P(A_1 \cap A_2 \cap A_3) = 0.02$

b. P (Doesn't like Football) = $P(1 - A_1) = 1 - .24 = .76$

c. P(Likes both football and basketball) = $P(A_1 \wedge A_2) = P(A_1) + P(A_2) - P(A_1 \cup A_2) = .24 + .18 - .3 = .12$

d. P (Likes football and Basketball but not swimming) = $P(A_1 \cap A_2 \cap A_3^c) = .12 - .02 = .1$

e. P (likes at most two sports) = $1 - P(A_1 \cap A_2 \cap A_3) = .98$

5.

a.

i. $P(A \wedge B \wedge C)$

ii. $P(A \wedge B^c \wedge C^c)$

iii. $P(A^c \wedge B^c \wedge C^c)$

iv. $P(A \vee B \vee C)$
v. $P(A \wedge B^c \wedge C^c) \vee P(A^c \wedge B \wedge C^c) \vee P(A^c \wedge B^c \wedge C)$

b. $P(A \vee B \vee C)$ is max if A B C are disjoint. In this case $P(A \vee B \vee C) = P(A) + P(B) + P(C) = .75$
$P(A \vee B \vee C)$ is min if C is a subset of B which is a subset of A. So $P(A \vee B \vee C) = .35$
c. $P(A \vee B) = P(A) + P(B) - P(A \wedge B) = .35 + .3 - .2 = .45$
d. $P(A \mid B) = P(A \wedge B)/ P(B) = 2/3$

6.
a. $d(k, 1) = 1$ for $k > 0$ because the only way to do this is to put k pumps in one tank. Likewise $d(k, n) = 0$ for $0\ n > k$ because for n tanks there should be at least n pumps, so there isn't a partition if $n > k$.
b. For the first tank you have K pumps and then for the next u have k -1 and so on. Which is equal to K! options
c. Since there is only one tank and more than one pump, then there are 2^k ways to decide whether or not to assign the k pump in the tank so $2^k -$ 1guarantees that.
d. Given j was set to the first tank, this leaves k - j to be assigned to n -1 tanks. J = 1,…, k- (n-1) so this is to make sure that there will be pumps left. If we give exactly j pumps to the first tank then there will be k choose j (d (k -j , n – 1), which after knowing j will give the equation provided.
e. 3.31×10^{13}
f. 1×10^6 assignments/sec * 3.15×10^7 sec/year $= 3.15 \times 10^{13}$ assignments per year
how long in years would it take the computer to analyze all possible assignments?
$3.15 \times 10^{13} / 3.31 \times 10^{31}$ ways assignments per year = 1.2.

```python

def nCr(n, r):
    f = math.factorial
    print(n)
    print(r)
    print(f(n) / f(r) / f(n-r))
    return f(n) / f(r) / f(n-r)


def code(k, n):
    sum = 0
    if n > k:

        return 0

    elif(n == k):

        return math.factorial(k)

    elif(n == 1 and k >= 0):

        return (2 ** k)-1

    else:
        for j in range(1, k - (n-1)+1):
            sum = sum + (nCr(k, j)) * code(k-j, n-1)
    return sum


print(code(15, 8))

```