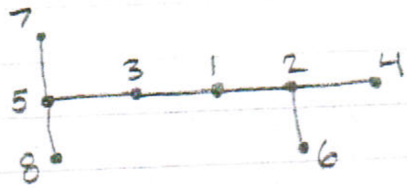
1.) i.) a.)



$i=1,\ v=4,\ s_1=2$

$i=2,\ v=6,\ s_2=2$

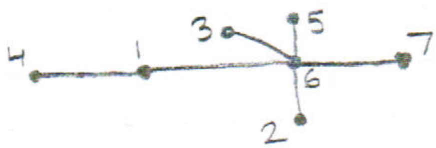
$i=3,\ v=2,\ s_3=1$

$i=4,\ v=1,\ s_4=3$

$i=5,\ v=3,\ s_5=5$

$i=6,\ v=7,\ s_6=5$ $\Rightarrow$ $\boxed{\langle 2,2,1,3,5,5\rangle}$

b.)



$i=1,\ v=2,\ s_1=6$

$i=2,\ v=3,\ s_2=6$

$i=3,\ v=4,\ s_3=1$

$i=4,\ v=1,\ s_4=6$

$i=5,\ v=5,\ s_5=6$ $\Rightarrow$ $\boxed{\langle 6,6,1,6,6\rangle}$

ii.) a.)

$i=1,$ $L=1,2,3,4,5,6,7,8$
$P=2,2,1,3,5,5$

$i=2,$ $L=1,2,3,5,6,7,8$
$P=2,1,3,5,5$

$i=3$ $L=1,2,3,5,7,8$
$P=1,3,5,5$

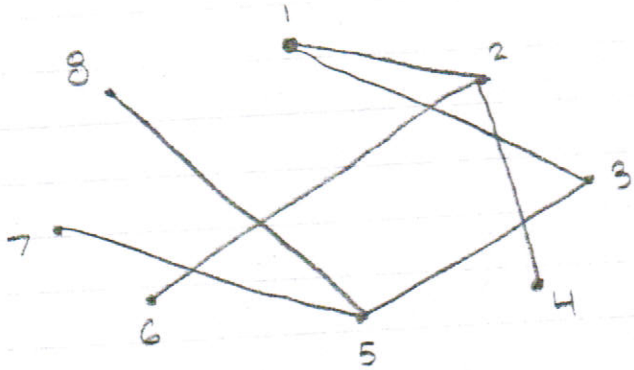
$i=4$ $L=1,3,5,7,8$
$P=3,5,5$

$i=5$ $L=3,5,7,8$
$P=5,5$

$i=6,$ $L=5,7,8$
$P=5$

finally we're left with $L=5,8$, so join 5 and 8

G:



b.)

i=1, L = 1, 2, 3, 4, 5, 6, 7
P = 6, 6, 1, 6, 6

i=2, L = 1, 3, 4, 5, 6, 7
P = 6, 1, 6, 6

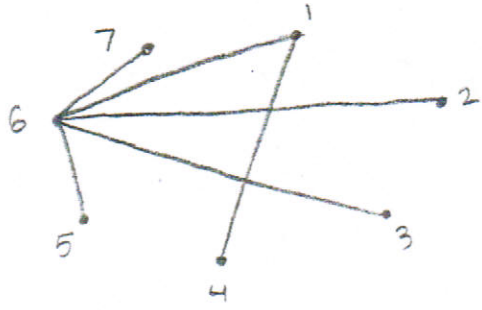
i=3, L = 1, 4, 5, 6, 7
P = 1, 6, 6

i=4, L = 1, 5, 6, 7
P = 6, 6

i=5, L = 5, 6, 7
P = 6

finally, we're left with L=6,7, so join 6 and 7

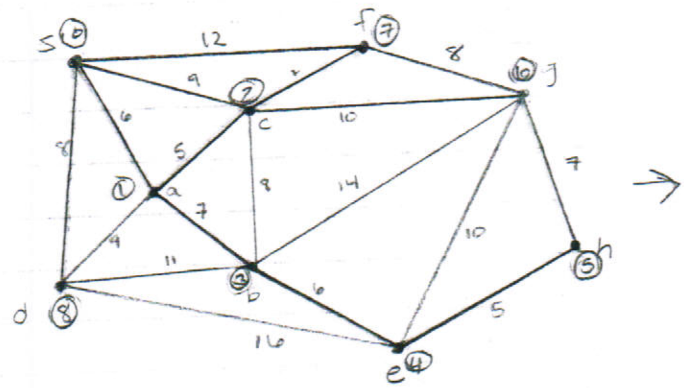
G:

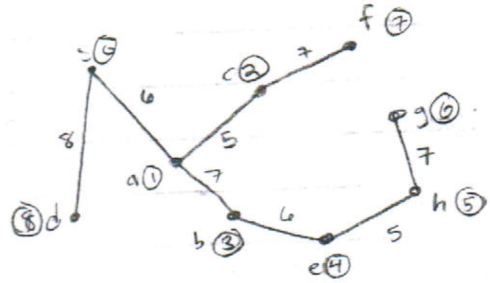


5/5

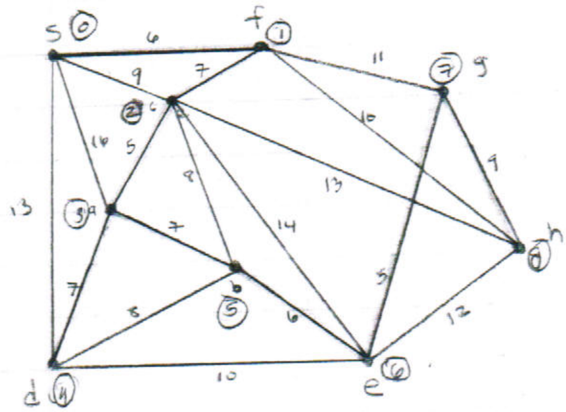
Result: The decoding process (demonstrated in ii.)) is the inverse of the encoding process (demonstrated in i.)).

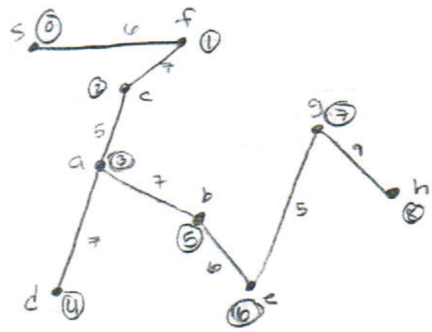
2)





$\sum$ weights = 8 + 6 + 7 + 6 + 5 + 7 + 5 + 7
= 51

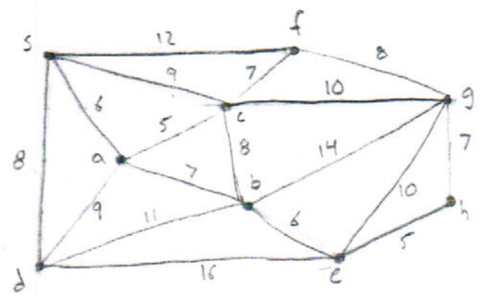




$\sum$ weights = 6 + 7 + 5 + 7 + 7 + 6 + 5 + 9
= 52.

5/5

3a)

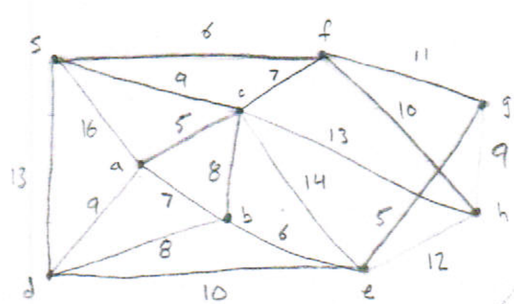


vertex: (discovery #, distance from s)

s: (0, 0)
a: (1, 6)
d: (2, 8)
c: (3, 9)
f: (4, 12)
b: (5, 13)
e: (6, 19)
g: (7, 19)
h: (8, 24)

5/5

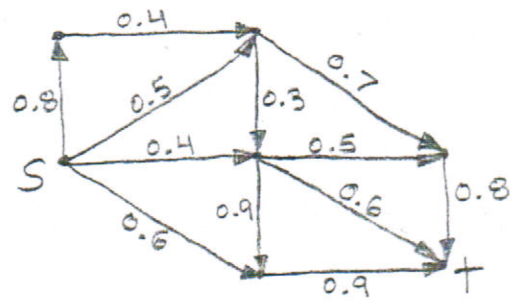
b)



vertex (discovery #, distance from s)

s: (0, 0)
f: (1, 6)
c: (2, 9)
d: (3, 13)
a: (4, 14)
h: (5, 16)
g: (6, 17)
b: (7, 17)
e: (8, 22)

4.) G:



edges = probability that the link does not fail (P_i)

we want to maximize the probability that the path from s to t does not fail.

So for a path $P_1 P_2 \cdots P_K$ from s to t

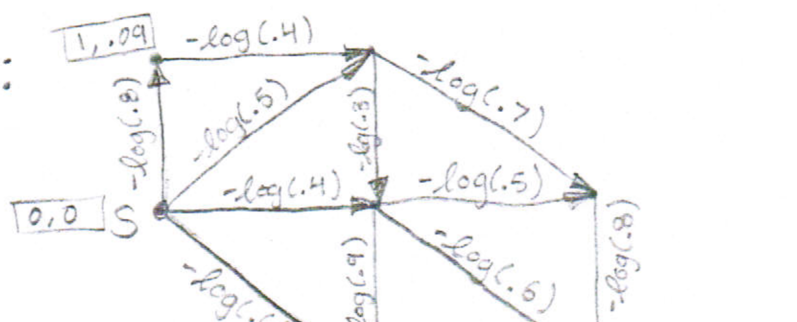
$$\log(P_1 P_2 \cdots P_K) = \log(P_1) + \log(P_2) + \cdots + \log(P_K)$$

we want to maximize $\log(P_1) + \cdots + \log(P_K)$, but if we negate this, we'll want to minimize $-\log(P_1) - \log(P_2) - \cdots - \log(P_K)$.

Therefore, if we transform each edge P_i in G using $-\log(P_i)$, we can apply Dijkstra's algorithm. (note that for $P_i < 1$, $-\log(P_i) > 0$ so we have no negative numbers)

5/5

G':



we can stop here!

($\boxed{a, b}$: a = discovery #, b = dist(s, p))

$\Rightarrow$ The most reliable path from s to t is P =