a.) The miss rate of the cache when running the code assuming that n is 32 is 25% because there are 3 hits for every miss (miss in every struct because each time you call something new into cache it is a miss) so 1024/(1024 + 3072) = 1024/4096 = 1/4 = 25% miss rate

b.) The miss rate of the cache when running the code assuming that n is 32 is 12.5% because there we have half as many misses as we did in part a because there is a miss for every other struct as opposed to a miss in every struct so 512/(512 + 3584) = 512/4096 = 12.5%

c.) g runs faster because there is better spacial locality, fewer misses and pulls from cache which is more efficient than pulling from memory. There are fewer misses in g because there are less calls of something new into cache. (view 7-2.c)