CMSC 351 HW1

Problem 1

slowSnatch $5 \cdot 2^n$ Tortoise: think slow. 1M/s

FastSnatch $500n^3$ Hare: think quick. 1T/s

a) (i) $\frac{500 \cdot 1000^3}{10^6} = 500(s)$

(ii) $\frac{500}{24 \cdot 60 \cdot 60} \approx 0.0058$ (Day)

b) (i) hare: $1,000,000,000,000 = 10^{12}$

$\frac{5 \cdot 2^{1000}}{10^{12}} = \frac{5 \cdot (2^{10})^{100}}{10^{12}} \approx \frac{5 \cdot 10^{300}}{10^{12}} = 5 \times 10^{288}$ (s)

(ii) $\frac{5 \times 10^{288}}{60 \cdot 60 \cdot 24 \cdot 100 \cdot 365} \approx 1.59 \times 10^{279}$ (c)

c) $\frac{5 \times 10^{288}}{500} = 10^{286}$ (times)

Problem 2:

a) $\Theta(n^3)$ Brute force

```
length ← 0
M ← 0
for(i:=0 to n)
  for(j:=0 to i+1)
    S ← 0 , L ← 0
    for(k=i to j)
      S ← S + A[i]
      L ← s.length
    end for
    M ← Max(M, S)
    length ← max(length, L)
end end
```

(b) $\Theta(n^2)$

```
Sum = [], temp = []
sum[0]
for(i=0 to n+1){
 sum[i] = sum[i] + sum[i-1]
}
 Max → min, max_Length → min
 for(i=0 , n){
   for(j=i+1, n){
     temp = sum[j] - sum[i-1]
   if(temp >= max && sum[j-1] >= max_length){
    max_ == sum
    max - length = j-1
    }
     return max-length
   end for
 end for
```

(c) $O(n)$

```
S[i] = Max(S[i-1]+A[i] ; 0)
M<0 , S[0] ← 0   length ← 0
length ← S.length
for('i=0 to n)
  S[i] ← Max( S[i-1] + A[i])
  M ← max( M, S[i])
  length ← M.length
end for
```