Team #:____ Name:_____

Absent: 8/28/07

1) Suppose that you have an $\theta(n^2)$ algorithm that required 10 seconds to run on a problem size of 1000. How long would you expect the algorithm to run on a problem size of 10,000?

2) Suppose that you have an $\theta(n^5)$ algorithm that required 10 seconds to run on a problem size of 1000. How long would you expect the algorithm to run on a problem size of 10,000?

3) Analyze the below algorithms to determine its theta notation, $\theta(\ )$.

```
index binary_search ( int n, const keytype S[ ], keytype x )   {
    index location, low, mid, high;
    low = 1;
    high = n;
    location = 0;
    while (location <= high && location == 0) {
        mid = (low + high) / 2;
        if (x == S[mid])
            location = mid;
        else if ( x < S[mid])
            high = mid - 1;
        else
            low = mid + 1;
    }  // end while
    return location
}
```

Team #:____ Name:_____

Absent: 8/28/07

```
void exchangesort (int n, keytype S[ ])  {
    index i, j;
    for ( i = 1; i <= n; i++) {
        for ( j = i + 1; j <= n; j++) {
            if ( S[j] < S[i] ) {
                temp = S[j];
                S[j] = S[i];
                S[i] = temp;
            } // end if
        } // end for j
    } // end for i
}
```

4) Analyze the below algorithm to determine its theta notation, $\theta(\ )$.

```
i := n
while i > 0 do
    for j = 1 to n do
        k := 1
        while k < i do
            < something of θ(1)>
            k := k * 2
        end while
    end for
    i := i / 2
end while
```