Worksheet - Algorithmic Complexity

Due No due date **Points** 0

Worksheet: Computational Complexity

1. Find the worst case upper bound complexity of the following functions.
(a) $f(n) = 64n$
(b) $f(n) = 1000000$
(c) $f(n) = 3n^2 logn + 6n + 100000$
(d) $f(x) = 5x^3 + logx + 66$
(e) $f(n) = 10log^2 n + 2n$
(f) $f(n, m) = 9n^2 + 99m + c$
2. Order the following functions by growth rate:

$$n, \sqrt{n}, n^{1.5}, n^2, nlogn, 3/n, nloglogn, 2^n, 6, n^3$$

3. What is the worst case upper bound on complexity of the following algorithm?

```
int getHandshakes(int n){
    int handshakes = 0;
    for (int i = 0; i < n; i++){
        for (int j = i+1; j < n; j++){
            handshakes++;
        }
    }
    return handshakes;
}
```

4. What is the worst case upper bound on complexity of the following function?

```
int getEvenIndexSum (int array[], int size){
    int sum = 0;
    for (int i = 0; i < size; i+=2){
        sum += array[i];
    }
    return sum;
}
```

5. What is the worst case upper bound on complexity of the following function?

```
void getPowers (int number) {
    for (int i = 1; i < number; i=i*2){
        cout << i << ", ";
    }
}
```

6. In what cases (best/average/worst) is the complexity of an algorithm important to the user of the algorithm?

Challenge Question:
1. Find two functions $f(n)$ and $g(n)$ such that neither $f(n) = O(g(n))$ nor $g(n) = O(f(n))$

(https://myuni.adelaide.edu.au/courses/54382/files/6299857/download?wrap=1)