Algorithms mandatory assignment 1

2 Runtimes

- First, it’s the function 3^n, this is the algorithm that has the most exponential growth, which means it’s the algorithm that takes the most time. After that we have x^3, which is almost as slow as the previously mentioned. Now there are 3 algorithms that share almost the same runtime, but there is a slightly difference between them. The slowest of them are 1.8^n, this one has a very similar curve as the first function. After 1.8^n we have n*(log2(n)) ^2, and after that we have n^2. All these mentioned algorithms have an exponential growth which means the runtime will increase drastically with the n. N is the next algorithm, and then we have sqrt(n), which is the second fastest. The fastest algorithm is log2(n) which won’t increase much runtime when the n increment.

- If the computers can compute 100 times faster in the future, sqrt(n) and log2(n) will still be almost the same runtime due to the runtime capping and not incrementing much with the change of n.

3 Min and Max

- Here I will write a pseudo code for finding the maximum and minimum value.

```
1. minimum = first object in the list
2. While(list has more objects){
3.     If(current object in list < minimum){
4.     minimum = current object}
5. } Return minimum
```

```
1. Maximum = first object in the list
2. While(list has more objects){
3.         If(current object in list > maximum){
4. maximum = current object}
5. }Return maximum
```

Line	Memory cost	How many times	Total
1.	1	1	1
2.	1	N	N
3.	1	N-1	N-1
4.	1	K	K
5.	-	-	0
			2N+k

Assuming operations, except the loop, takes 1 cost the total cost of the program 2N+K.

Worst case for n=5:

Line	Memory cost	How many times	Total
1.	1	1	1
2.	1	5	5
3.	1	4	4
4.	1	4	4
5.	-	-	0
			14

Worst case the program must change the minimum object every time, which means the n is 5 and the k is 4. This gives us a total cost of 15. Worst case for maximum means that the list is sorted ascending and sorted descending for the minimum program. Best case for these algorithms means that the k is reduced to 0, which means sorted ascending for minimum and descending for the maximum algorithm. Then the total cost of these algorithms would be 10.

4 Binary counter

The changing in 0’s and 1’s is 1, 2,1,3,1,2,1, n. For each time you must change all the previous 1’s and change one 0 you the cost is n.