Bowen Wang
301267523
CMPT 125
Assignment 1

Assignment 1

1. a) The loop body of i runs in O(1), and executed O(n) times, the loop body of j runs in O(n), and executed O(1) times, the loop body of k runs in O(1), and executed O (1) times, so the total running time is O(1)*O(1)*O(n)=O(n).

   b) The loop body of i runs in O(1), and executed O(n-1) times,

   the loop body of j executed O ($\log_2 n$), the loop body of k executed O($\log_2 n$), so the total running time is O(n-1)* O (2$\log_2 n$)=O(n*$\log_2 n$)

   c) The loop body of j runs in O(1), and executed O(n) times in the inner while loop, the loop body of i runs O(n) times in the outer while loop, so the total running time is O(n)*O(n)=O(n^2)

   d) The inner loop of i runs (n/2) times, and the outer loop of i runs (n/2) times, so the total running time is O(n).
2. a)
   The total times of compare is: n-1, so T(n)=O(n).
   b)
   The worst case is there exist one couple and Bob finds them at the final comparison, the total times of comparison is: (n-1) * (n-2) * … * 2 * 1=n(n-1)/2, so T(n)=O(n^2)
3. a) 257
   b) 51
   c) 336
   d) 17
4. a) O (n^3)
   b) O (n^2)
   c) O (n*logn)

Bowen Wang
301267523
CMPT 125
Assignment 1

d) O (n^2)

5. input array[]
   int len is the length of array[]
   int new_arr[len]
   for int i in range len
   if array[i] != every element in new_arr, replace the value in new_arr with the selected element
   return new_arr

6.

```
int bestrewards(int array[],int len){
        int i=0, j, test, max=array[0];
        while (i<len){
                test =0;
                if (array[i]<0){
                        i++;
                }else {
                        for (j=i, j<len, j++){
                                test += array[j];
                                if (test>max){
                                        max = test;
                                }
                        }
                }
        }
        return max;
}
```