COMP9101 Ass02
Name: Zanning Wang
zID: z5224151

Q1:

In order to satisfy all the requirement, we should fill the CD with the songs as many as possible, as long as the total time not exceed the constraint m. the sudo-code show below:

```
1  song_n = [n1, n2, n3,...,ni];
2  let counter_CD = 0;
3  let Max_time = m;
4  let current = 0;
5  for (let i=0; i<= song_n.length; i++) {
6    if (current <= Max_time) {
7      current += song_n[i]
8    } else {
9      current = 0;
10     counter_CD += 1
11   }
12 }
13 return counter_CD
```

For line 5-12, we assign each song to corresponding CD according to the requirement, and the time complexity is O(n);
Therefore, the total time complexity is O(n);
In order to prove this algorithm is right, suppose there is another algorithm, which can make each CDs contain more songs than current algorithm, the total time of current CD must exceed the time m, this algorithm does not satisfy the requirement A, and we are also not allowed to split songs, therefore we can only put the song which exceed the requirement to the next CD. After repeating that in each CD below, we will get the same algorithm as we mentioned before.