Longest Consecutive Sequence

```
class Solution {
public:
    int longestConsecutive(vector<int>& nums) {
      int i;
      int j;
      unordered_map<int,int> map;
      unordered_map<int,int> map1;
      int awn = 0;

      for (i = 0;i<nums.size();i++){
        if (!map.count(nums[i])) map[nums[i]] += 1;
      }

      for (i = 0;i<nums.size();i++){
        if (!map.count(nums[i]-1)){
          map1[nums[i]] += 1;
        }
      }

      for (auto j : map1){
        for (i = 0;i<nums.size();i++){
          if (!map.count(j.first + i)) break;
          else map[j.first] = i;
        }
      }

      for (auto i: map){
        if (i.second +1 > awn) awn = i.second+1;
      }

      return awn;
    }
};
```

- Working solution
    - sequence have a start and an end
    - starts have no numbers before them
    - find all numbers that have no numbers before them
        - these are all starts of sequences
    - count through each sequence and find out which one is the longest
    - how would i do this as a human and let me get a computer to do it