**TRAINING PRACTICE**

1. Fib series  
   Using two variables
2. class Solution {
3. public:
4. int fib(int n) {
5. int a=0;
6. int b=1;
7. if(n==0) return a;
8. int i=2;
9. while(i<=n){
10. b=a+b;
11. a=b-a;
12. i+=1;
13. }
14. return b;
15. }
16. };

Using Recurssion

class Solution {

public:

    int rec(int n){

        if(n==0) return 0;

        if(n==1) return 1;

        return rec(n-2)+rec(n-1);

    }

    int fib(int n) {

        return rec(n);

    }

};

1. Running sum of 1d array

class Solution {

public:

    vector<int> runningSum(vector<int>& nums) {

        vector <int> s;

        s.push\_back(nums[0]);

        for(int i=1;i<nums.size();i++){

            s.push\_back(nums[i]+s[i-1]);

        }

        return s;

    }

};

Manipulating the same array  
  
class Solution {

public:

    vector<int> runningSum(vector<int>& nums) {

        for(int i=1;i<nums.size();i++){

            nums[i]+=nums[i-1];

        }

        return nums;

    }

};

1. Sort Integers by number of bits

class Solution {

public:

    vector<int> sortByBits(vector<int>& arr) {

        map<int, vector<int> > d;

        int count,num;

        for(auto n: arr){

            count=0;

            num=n;

            while(num!=0){

                if (num%2==1) count++;

                num/=2;

            }

            if(d.find(count)!= d.end())

                d[count].push\_back(n);

            else

                d[count]= vector<int> {n};

        }

        vector<int> ans;

        for(auto [k,v]: d){

            sort(v.begin(),v.end());

            ans.insert(ans.end(),v.begin(),v.end());

        }

        return ans;

    }

};

Using vector pair

class Solution {

public:

    vector<int> sortByBits(vector<int>& arr) {

      vector < pair <int,int>> ans;

       int count,num;

       for (auto n: arr){

            count=0;

            num=n;

            while(num!=0){

                if(num%2==1)count++;

                num/=2;

            }

            ans.push\_back({n,count});

       }

       sort(ans.begin(),ans.end(),[](pair<int,int>a, pair<int,int>b){

        if(a.second==b.second) return a.first<b.first;

        return a.second<b.second;

       });

       vector<int> res;

       for(auto a: ans)

           res.push\_back(a.first);

       return res;

    }

};