

1. First Problem :

```
#include <bits/stdc++.h>

using namespace std;

bool isPalindrome(string str)
{
    for (int i = 0; i < str.length() / 2; i++)
    {
        if (str[i] != str[str.length() - 1 - i])
        {
            return false;
        }
    }
    return true;
}

string parity(int num)
{
    if (num % 2 == 0)
    {
        return "even";
    }
    else
    {
        return "odd";
    }
}

int main()
{
    ifstream inputFile("input.txt");
    ofstream outputFile("output.txt");
    ofstream recordFile("record.txt");

    float number;
    string s;

    int totalOddParity = 0;
    int totalEvenparity = 0;
    int noparity = 0;
    int totlaPalindrome = 0;
    int totalNonPalindrome = 0;
```

```

while (inputFile >> number >> s)
{
    int x = floor(number);
    if (x != number)
    {
        outputFile << number << " cannot have parity"
            << " and ";
        noparity++;
    }
    else
    {
        outputFile << number << " has " << parity(number) << " parity"
            << " and ";
        if (parity(number) == "even")
        {
            totalEvenparity++;
        }
        else
        {
            totalOddParity++;
        }
    }
    if (isPalindrome(s))
    {
        outputFile << s << " is a palindrome" << endl;
        totlaPalindrome++;
    }
    else
    {
        outputFile << s << " is not a palindrome" << endl;
        totalNonPalindrome++;
    }
}

recordFile << "Percentage of Odd Parity: " << (100 * totalOddParity) / (totalOddParity +
totalEvenparity + noparity) << "%" << endl;
recordFile << "Percentage of Even Parity: " << (100 * totalEvenparity) / (totalOddParity +
totalEvenparity + noparity) << "%" << endl;
recordFile << "Percentage of No Parity: " << (100 * noparity) / (totalOddParity +
totalEvenparity + noparity) << "%" << endl;
recordFile << "Percentage of Palindrome: " << (100 * totlaPalindrome) / (totlaPalindrome +
+totalNonPalindrome) << "%" << endl;
recordFile << "Percentage of Odd Parity: " << (100 * totalNonPalindrome)
/ (totlaPalindrome + +totalNonPalindrome) << "%" << endl;

inputFile.close();

```

```

outputFile.close();
recordFile.close();
}

```

```

test.cpp
input.txt
1 1 madam
2 2 apple
3 3.6 racecar
4 89 parrot
5 45.2 github

output.txt
1 1 has odd parity and madam is a palindrome
2 2 has even parity and apple is not a palindrome
3 3.6 cannot have parity and racecar is a palindrome
4 89 has odd parity and parrot is not a palindrome
5 45.2 cannot have parity and github is not a palindrome
6

record.txt
1 Percentage of Odd Parity: 40%
2 Percentage of Even Parity: 20%
3 Percentage of No Parity: 40%
4 Percentage of Palindrome: 40%
5 Percentage of Odd Parity: 60%
6

```

2. Second Problem

```

#include<bits/stdc++.h>
using namespace std;

int fibonacci(int a) {
    if((a==1)||(a==0)) {
        return a;
    }else {

```

```

        return(fibonacci(a-1)+fibonacci(a-2));
    }
}
int main()
{
    int n , i=0;
    cin >> n;
    cout << "Fibonacci Series : " << endl;
    while(i < n) {
        cout << " " << fibonacci(i);
        i++;
    }
    return 0;
}

```

3. Third Problem :

```

#include <bits/stdc++.h>
using namespace std;

int binarySearch(int arr[], int l, int h, int x)
{
    if (l > h)
    {
        return -1;
    }

    int mid = (l + h) / 2;

    if (arr[mid] == x)
    {
        return mid;
    }
    else if (arr[mid] < x)
    {
        return binarySearch(arr, mid + 1, h, x);
    }
}

```

```

    }
    else
    {
        return binarySearch(arr, l, mid - 1, x);
    }
}

int main() {
    int arr[] = {0, 2, 4, 5, 6, 7, 8, 15, 17, 19, 21, 23, 25, 27, 29};

    int target;
    cin >> target;

    int index = binarySearch(arr, 0, 15, target);

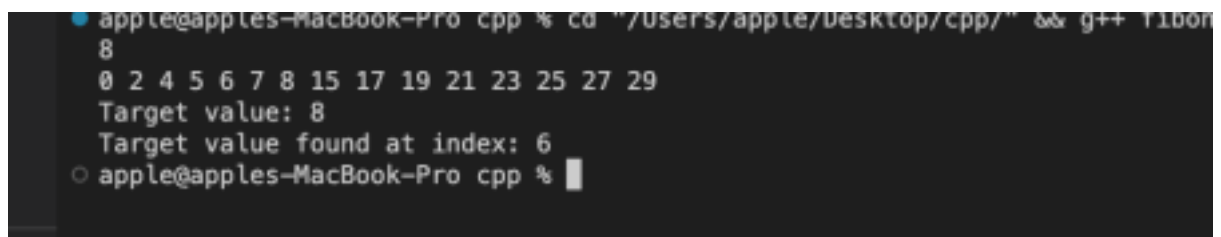
    for (int i = 0; i < 15; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;

    cout << "Target value: " << target << endl;

    if (index == -1) {
        cout << "Target value not found." << endl;
    } else {
        cout << "Target value found at index: " << index << endl;
    }

    return 0;
}

```



```

apple@apples-MacBook-Pro cpp % cd "/Users/apple/Desktop/cpp/" && g++ Fibonacci.cpp -std=c++11 -o Fibonacci
8
0 2 4 5 6 7 8 15 17 19 21 23 25 27 29
Target value: 8
Target value found at index: 6
apple@apples-MacBook-Pro cpp %

```