Задание 2. Нижние оценки и числа Фибоначчи.

1. Воспользуемся биномом Ньютона: $(a+b)^n = \sum\limits_{k=0}^{n} \binom{n}{k} a^k b^{n-k}$. Видно, что если подставить $a = b = 1$, получим $f(n)$. Отсюда $f(n) = 2^n = \Theta(2^n)$.

Ответ: $\Theta(2^n)$.

2. Алгоритм: На каждом шаге делим кучу пополам, если монет нечетное количество, оставляем одну. Взвешиваем кучи, повторяем шаг для более легкой. Если кучи равны по весу, значит монет было нечетное количество, и та, которую оставили лежать — фальшивая. Псевдокод:

```
function FindFalseCoin(coins)
    if #coins == 1 then
        return coins
    else if #coins == 2n then
        #coinsLeft = n, #coinsRight = n
        if mass(coinsLeft) < mass(coinsRight) then
            FindFalseCoin(coinsLeft)
        else
            FindFalseCoin(coinsRight)
        end if
    else if #coins == 2n + 1 then
        #coinsLeft = n, #coinsRight = n, singleCoin
        if mass(coinsLeft) < mass(coinsRight) then
            FindFalseCoin(coinsLeft)
        else if mass(coinsLeft) > mass(coinsRight) then
            FindFalseCoin(coinsRight)
        else
            return singleCoin
        end if
    end if
end function
```

Корректность: На каждом шаге алгоритм применяется к той куче, в которой лежит фальшивая монета. После каждого шага размер кучи строго уменьшается: $2n \to n, 2n+1 \to n$, если не происходит преждевременной остановки в случае нечетного числа монет. Значит, за конечное число шагов алгоритм будет применен к куче, состоящей из одной монеты и завершится, выдав фальшивую.

Асимптотика: За один шаг размер кучи сокращается вдвое. Пусть $2^{k-1} < \#\text{Монет} = n \le 2^k$. Отсюда $k-1 < \log n \le k$. Обозначим $T(n)$ — количество шагов алгоритма для n монет. Тогда $k-1 = T(2^{k-1}) \le T(n) \le T(2^k) = k$. Т. е. $T(n) = \Theta(\log n)$.

3. Докажем, что задача нахождения фальшивой монеты имеет асимптотику $\log n$.
Предположим, что дано 3^k монет, и мы пронумеруем их от 0 до $22\ldots2$ в троичной системе. Тогда для нахождения фальшивой монеты необходимо узнать ее номер. Чтобы узнать одну цифру номера, необходимо определить $\frac{2}{3}$ монет, как настоящие. Для этого надо разделить монеты на 3 кучи и взвесить две. Заметим, что больше одной цифры узнать за взвешивание нельзя, так как если в каждой куче, которые взвешиваем будет более трети монет, то можем опознать одну из них и оставшуюся, т. е. менее $\frac{2}{3}$, а если взвешивать кучки менее трети монет, то в случае их равенства, как настоящие мы определим менее $\frac{2}{3}$. Итак, каждое взвешивание дает не более одной цифры, всего цифр k, т. е. нужно k взвешиваний. Остается заметить, что если количество монет находится между 3^{k-1} и 3^k, то цифр для записи нужно $k = \lceil \log_3 n \rceil$.

4. $13 = 1101_2$.

1. $b = 1$: $(7^2) \cdot 7 \equiv 343 \equiv 343 - 334 \equiv 9 \mod 167$,
2. $b = 0$: $(7^3)^2 \equiv 9^2 \equiv 81 \mod 167$,
3. $b = 1$: $(7^6)^2 \cdot 7 \equiv 81^2 \cdot 7 \equiv 48 \cdot 7 \equiv 2 \mod 167$.

Ответ: 2.