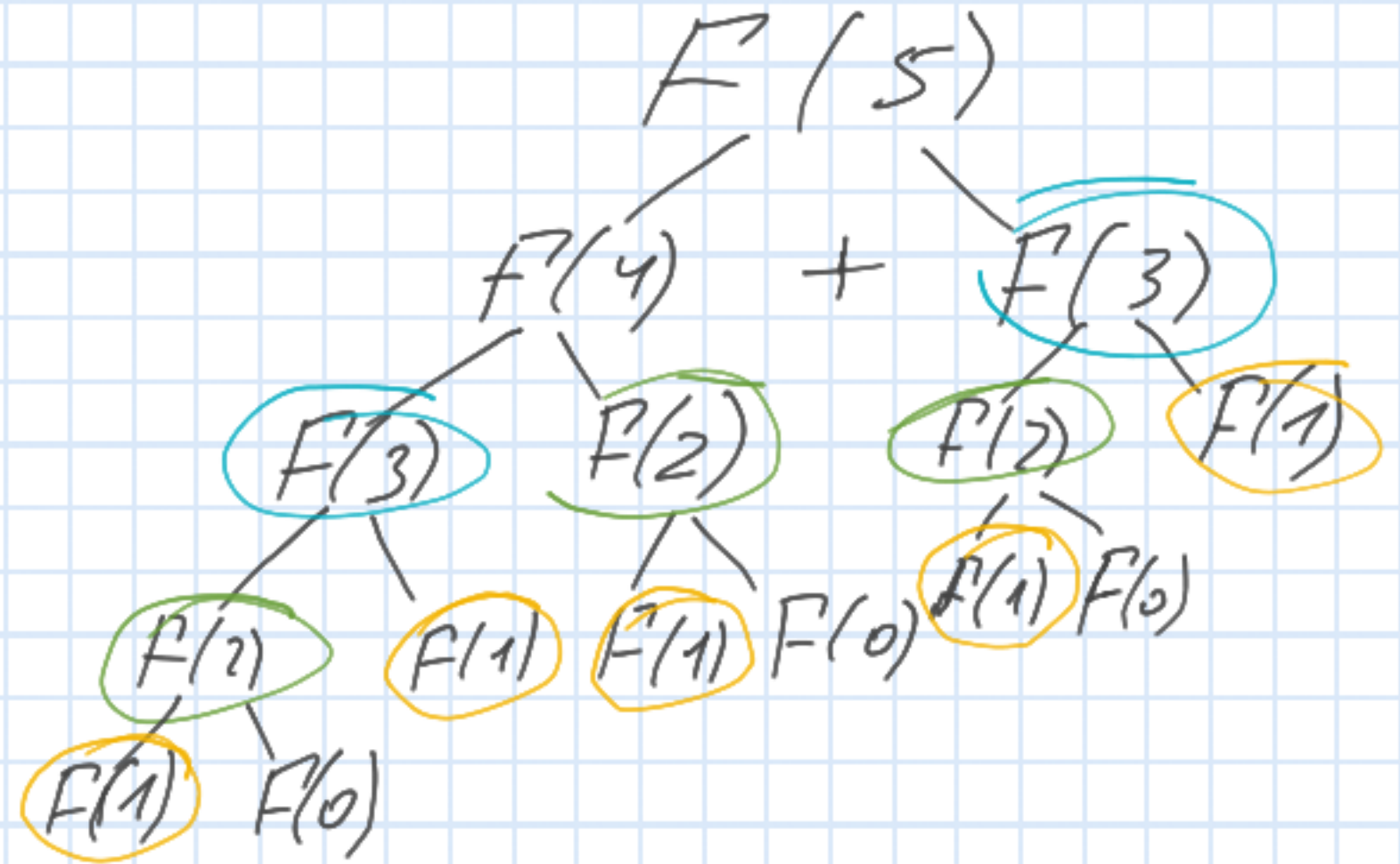


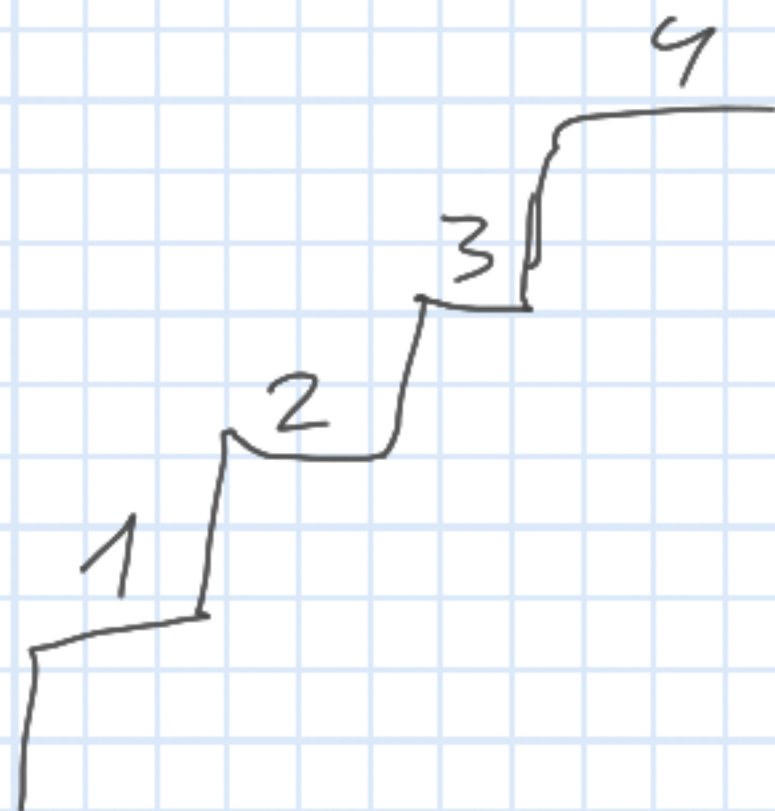
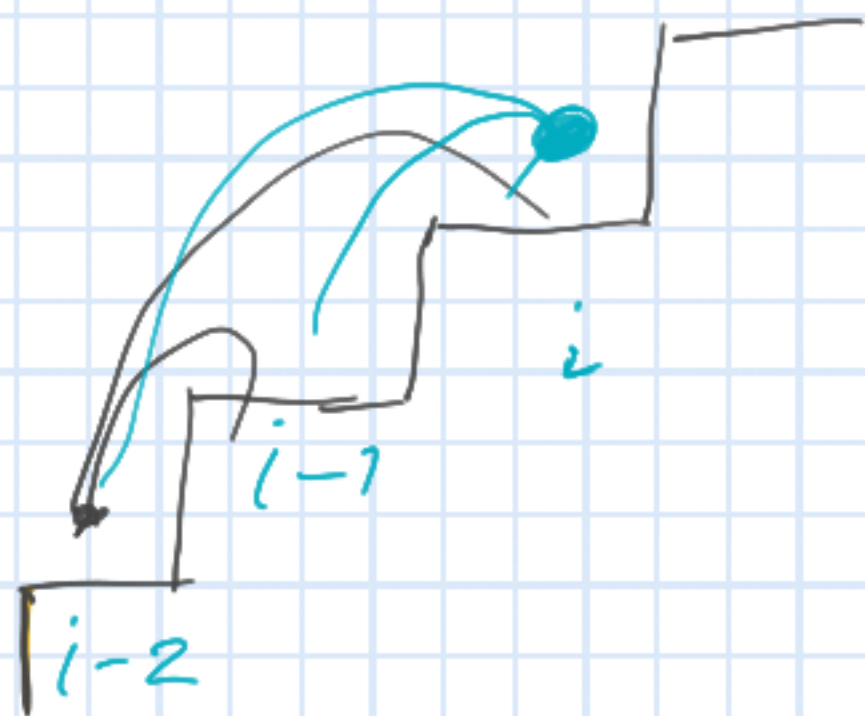
Динамическое программирование

1) Числа Фибоначчи:
 $F(0)=0$, $F(1)=1 \leftarrow$
 $F(n) = F(n-1) + F(n-2)$

0, 1, 1, 2, 3, 5, ...



Платная лестница

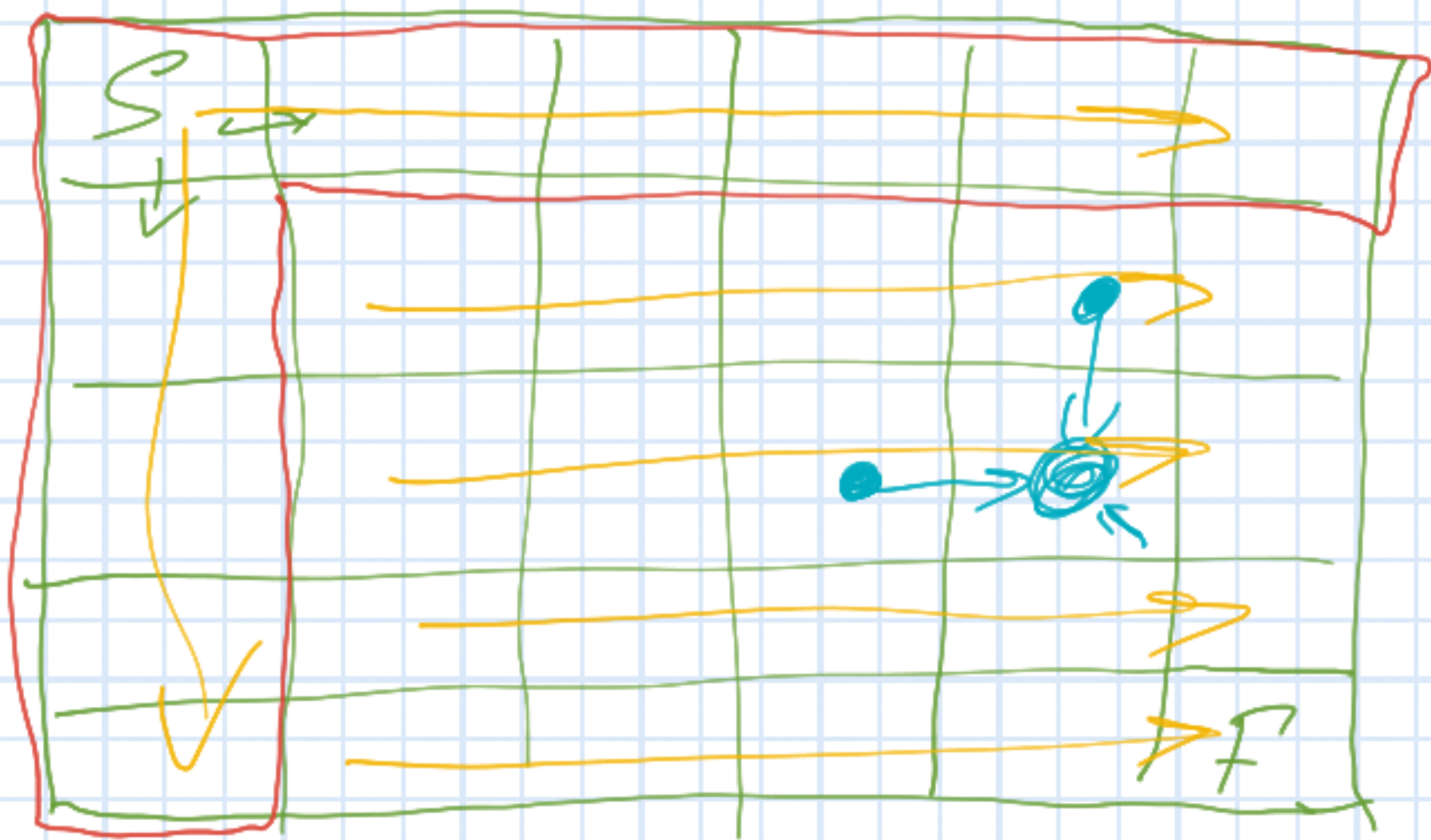


n - ступеней
 a_1, \dots, a_n - стоимость ступеней

$dp(i)$ - мин. стоимость до i -й ступеньки

$$dp(i) = \min(dp(i-1), dp(i-2)) + a_i$$

Робот-сборщик мусора.



$$\begin{cases} dp[i][j] = \max(dp[i][j-1], dp[i-1][j]) + \text{coins}[i][j] \\ dp[i][0] = dp[i-1][0] + \text{coins}[i][0] \\ dp[0][0] = \text{coins}[0][0] \end{cases}$$

Кафе

Около Петинго университета недавно открылось новое кафе, в котором действует следующая система скидок: при каждой покупке более чем на 100 рублей покупатель получает купон, дающий право на один бесплатный обед (при покупке на сумму 100 рублей и меньше такой купон покупатель не получает).

Однажды Пете на глаза попался прейскурант на ближайшие N дней. Внимательно его изучив, он решил, что будет обедать в этом кафе все N дней, причем каждый день он будет покупать в кафе ровно один обед. Однако стипендия у Пети небольшая, и поэтому он хочет по максимуму использовать предоставляемую систему скидок так, чтобы его суммарные затраты были минимальны. Требуется найти минимально возможную суммарную стоимость обедов и номера дней, в которые Пете следует воспользоваться купонами.

Входные данные

В первой строке входного файла записано целое число N ($0 \leq N \leq 100$). В каждой из последующих N строк записано одно целое число, обозначающее стоимость обеда в рублях на соответствующий день. Стоимость — неотрицательное целое число, не превосходящее 300.

Выходные данные

В первой строке выдайте минимальную возможную суммарную стоимость обедов. Во второй строке выдайте два числа K_1 и K_2 — количество купонов, которые останутся неиспользованными у Пети после этих N дней и количество использованных им купонов соответственно.

В последующих K_2 строках выдайте в возрастающем порядке номера дней, когда Пете следует воспользоваться купонами. Если существует несколько решений с минимальной суммарной стоимостью, то выдайте то из них, в котором значение K_1 максимально (на случай, если Петя когда-нибудь ещё решит заглянуть в это кафе). Если таких решений несколько, выведите любое из них.

	150	80	1000
	0	1	2
1	$+\infty$	150	$+\infty$
2	150	230	$+\infty$
3	230	1150	1230

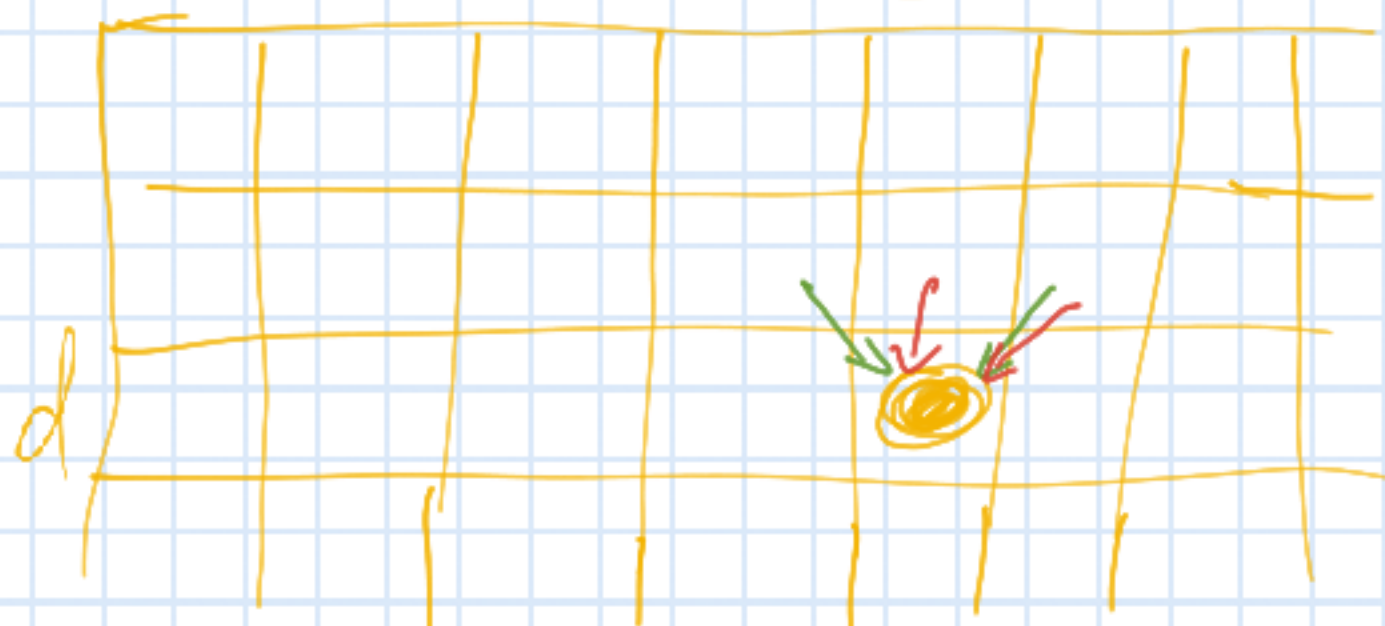
off top

```
int n; (unsigned long long) 1 <= n
1 <= n
int int int
1 <= n
```

1s	1us	1p
1u		1.0
1e	1ul	1ed
1ll	1ull	

$dp[d][c]$

мин сумма, которую мы потратим на d-й день, при усл., что обоимее с купонов.



> 100

≤ 100

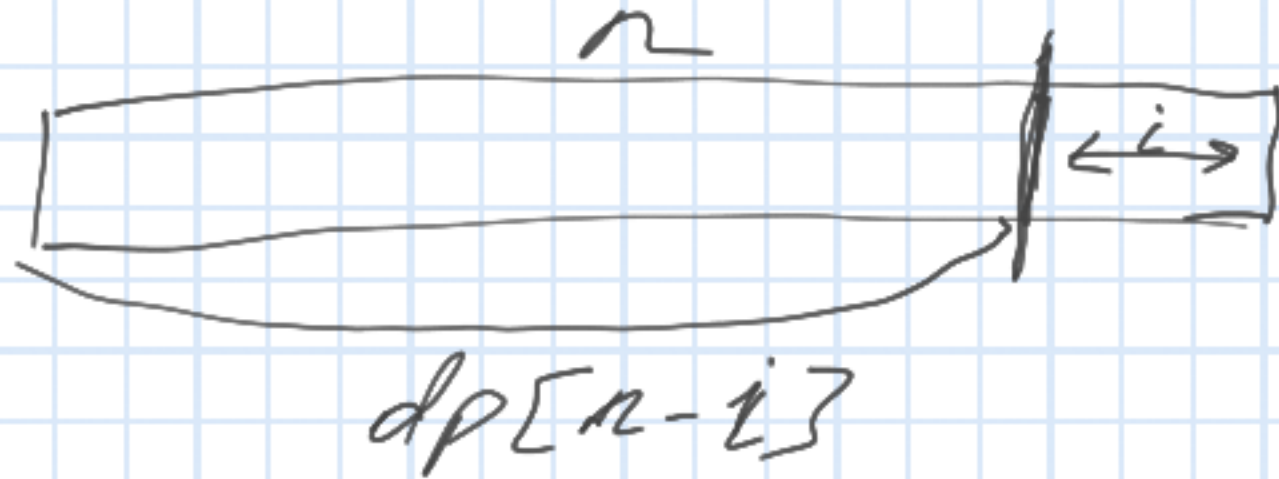
$dp[d][c] =$

if cost[d] > 100 then
 $\min(dp[d-1][c-1] + cost[d], dp[d-1][c+1])$
 else
 $\min(dp[d-1][c] + cost[d], dp[d-1][c+1])$

Задача о проезде

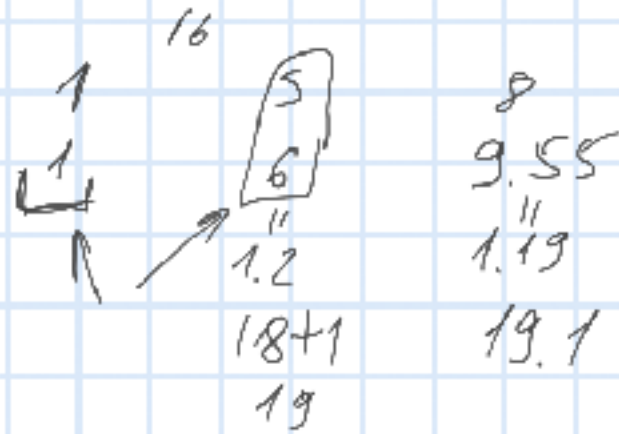
n — длина бревна

a_1, \dots, a_n ; a_i — стоимость бревна
длины i .



n — объём проката

a_1, \dots, a_n ; a_i — цена объёма
объёмом i



$$dp[k] = \max_i (a_i + dp[k-i])$$

```
for (k=1; k<=n; k++) {  
    dp[k] = a[1] + dp[k-1];
```

```
    for (i=2; i<=k; i++)
```

```
        dp[k] = max(dp[k], a[i] + dp[k-i])
```

```
}
```


V - обобщенная
 a_1, \dots, a_n

V_1, \dots, V_n $V_{i-1} = 4$

	0	1	2	\dots	V
0	0	$-\infty$	$-\infty$	$-\infty$	$-\infty$
1	$-\infty$	$-\infty$	$-\infty$	a_1	$-\infty$
2		a_2	a_1	$a_1 + a_2$	
3	a_3	$a_2 + a_3$	$a_1 + a_3$	$a_1 + a_2 + a_3$	
\vdots					
n					

$\max(a_1, a_2)$

$0 \mid -\infty \mid -\infty \dots \mid -\infty$