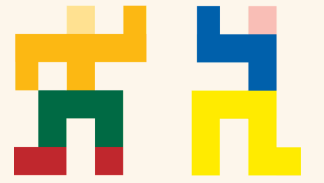


ELEVATOR



STATEMENT



You have an elevator that goes up from floor 0 to floor N ($N = 60$) with buttons for each floor and on each floor there is an unknown value v_i . This task is of encode-decode type. On each floor the lift goes you can press any unpressed buttons for the upper floors. On the rooftop you have to decode some of the values v_i by knowing only the pressed buttons. The elevator goes to the floors with pressed buttons. We will denote with K the number of decoded values.

Constraints: $0 \leq v_i \leq 3$

Special subtask, best $K = 60$;

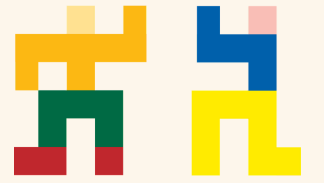
For $v_i \leq 1$, best $K = 40$.

For $v_i \leq 2$, best $K = 30$.

For $v_i \leq 3$, best $K = 25$.



ELEVATOR



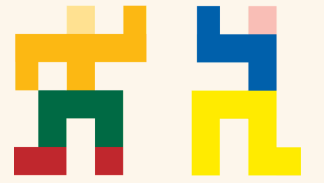
Naive strategy

We can split the floors in two types - for the values to encode and for making the encoding.

For $v_i \leq 1$, we could encode $K = 30$ values $v_0, \dots, v_{29}$ in $v_{30}, \dots, v_{59}$ (if the button for floor $j \geq 30$ is pressed that would mean that v_{j-30} is 1 and otherwise 0). With a little bit of more effort we could even encode $K = 31$ values.



ELEVATOR



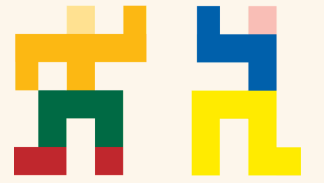
Naive strategy

For $v_i \leq 3$, we could encode $K = 20$ values $v_0, \dots, v_{19}$ in $v_{20}, \dots, v_{59}$ using 2 floors for each value by looking at the binary representation of v_i .

Using this strategy we get 46.5 points from subtask 1 to 4 ($7.5 + 15 + 10 + 14$).



ELEVATOR

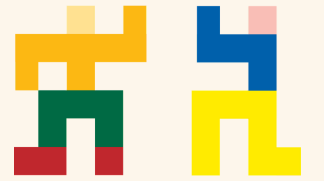


Better idea

Splitting the floors in two types is obviously not very efficient and we cannot make a better solution with that approach. The better idea is to not have such a split and use a floor potentially for both purposes. We will try to encode the values using the elevator movements and then make a "replay" of this during the decoding part.



ELEVATOR



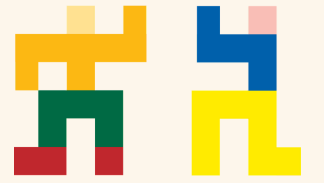
Subtask 1 - $K = 60$

Although it is the easiest subtask, for full points this subtask forces us to use the better idea. There are no consecutive 0s so if we see a zero we know the next value is 1. Thus, we can encode the blocks of 1s in the following way:

	0	1	2	3	4	5	6	7	8	9	10	11	...
v_i	1	1	0	1	1	1	0	1	0	1	1	0	...
p_i	-	1	1	0	1	1	1	0	1	0	1	1	...



ELEVATOR



Subtask 3 - $K = 30$

We would use floors (1, 2) for encoding the first recovered value, (3, 4) for encoding the second recovered value and etc. To be able to encode 30 values we would need something like the following encoding that makes sure we will be on at least one floor for each of these 30 groups:

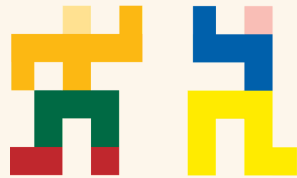
$$v_i = 0 \rightarrow p_{2i+1}, p_{2i+2} = 10$$

$$v_i = 1 \rightarrow p_{2i+1}, p_{2i+2} = 01$$

$$v_i = 2 \rightarrow p_{2i+1}, p_{2i+2} = 11$$



ELEVATOR



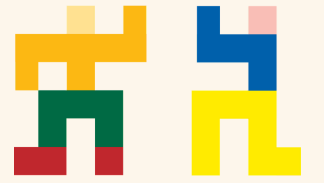
Subtask 3 - $K = 30$

Example

	0	1	2	3	4	5	6	7	8	9	10	11	12	...
v_i	1	2	0	1	1	2	0	0	0	1	1	0	2	...
p_i	-	1	0	1	1	1	0	1	0	1	1	0	1	...



ELEVATOR



Subtask 3 - $K = 30$

We could make this idea a little bit cleaner if we only encode one floor from a group like that:

1.[00-00] $\rightarrow$ [01-02] - 1 encoded

2.[01-02] $\rightarrow$ [03-04] - 1 encoded

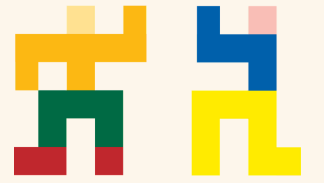
3.[03-04] $\rightarrow$ [05-06] - 1 encoded

...

30.[57-58] $\rightarrow$ [59-60] - 1 encoded



ELEVATOR



Subtask 2 - K = 40

We will do a similar thing and encode two floors in a group of three floors. To be able to encode 40 values we would need something like the following encoding that makes sure we will be on at least two floors for each of these groups:

$$v_i, v_j = 00 \rightarrow p_{3h+2}, p_{3h+3}, p_{3h+4} = 011$$

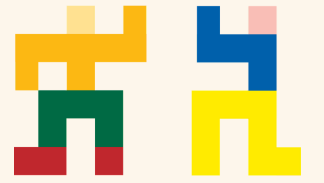
$$v_i, v_j = 01 \rightarrow p_{3h+2}, p_{3h+3}, p_{3h+4} = 101$$

$$v_i, v_j = 10 \rightarrow p_{3h+2}, p_{3h+3}, p_{3h+4} = 110$$

$$v_i, v_j = 11 \rightarrow p_{3h+2}, p_{3h+3}, p_{3h+4} = 111$$



ELEVATOR



Subtask 2 - K = 40

Then the encoding would look like this:

1.[00-01] $\rightarrow$ [02-04] - 2 encoded

2.[02-04] $\rightarrow$ [05-07] - 2 encoded

3.[05-07] $\rightarrow$ [08-10] - 2 encoded

...

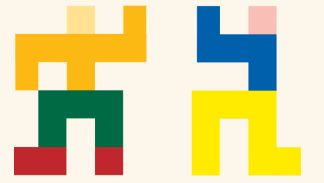
19.[53-55] $\rightarrow$ [56-58] - 2 encoded

20'.[56-58]₁ $\rightarrow$ 59 - 1 encoded

20''.[56-58]₂ $\rightarrow$ 60 - 1 encoded



ELEVATOR



Subtask 4 - $K = 24$

We will do a similar thing and encode two floors in a group of five floors. To be able to encode 24 values we would need encoding that makes sure we will be to at least two floors of each of these groups. Then the encoding would be:

1. $[00-01] \rightarrow [02-06]$ - 2 encoded

2. $[02-06] \rightarrow [07-11]$ - 2 encoded

3. $[07-11] \rightarrow [12-17]$ - 2 encoded

...

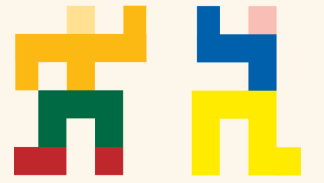
19. $[47-51] \rightarrow [52-56]$ - 2 encoded

20'. $[52-56]_1 \rightarrow [57-58]$ - 1 encoded

20''. $[52-56]_2 \rightarrow [59-60]$ - 1 encoded



ELEVATOR



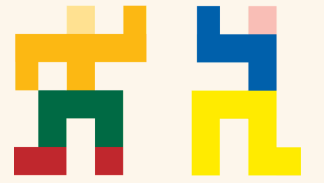
Subtask 4 - K = 25

For full points we will encode two floors in a group of five floors ensuring that at least three buttons are pressed in a group, then all other floors - with the direct binary encoding in rest of the floors:

- 1.[00-01] $\rightarrow$ [02-06] - 2 encoded
- 2'.[02-06]_{1,2} $\rightarrow$ [07-11] - 2 encoded
- 2''.[02-06]₃ $\rightarrow$ [12-16] - 1 encoded
- 3'.[07-11]₁ $\rightarrow$ [12-16] - 1 encoded
- 3''.[07-11]_{2,3} $\rightarrow$ [17-21] - 2 encoded
- 4'.[12-16]_{1,2} $\rightarrow$ [22-26] - 2 encoded
- 4''.[12-16]₃ $\rightarrow$ [27-31] - 1 encoded
- 5'.[17-21]₁ $\rightarrow$ [27-31] - 1 encoded
- 5''.[17-21]_{2,3} $\rightarrow$ [27-31] - 2 encoded



ELEVATOR



Subtask 4 - K = 25

$[22-26]_1 \rightarrow [37-38] - 1$ encoded

$[22-26]_2 \rightarrow [39-40] - 1$ encoded

$[22-26]_3 \rightarrow [41-42] - 1$ encoded

$[27-31]_1 \rightarrow [43-44] - 1$ encoded

$[27-31]_2 \rightarrow [45-46] - 1$ encoded

$[27-31]_3 \rightarrow [47-48] - 1$ encoded

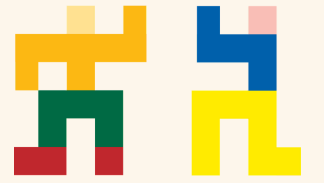
$[32-36]_1 \rightarrow [49-50] - 1$ encoded

$[32-36]_2 \rightarrow [51-52] - 1$ encoded

$[32-36]_3 \rightarrow [53-54] - 1$ encoded



ELEVATOR



Subtask 4 - K = 25

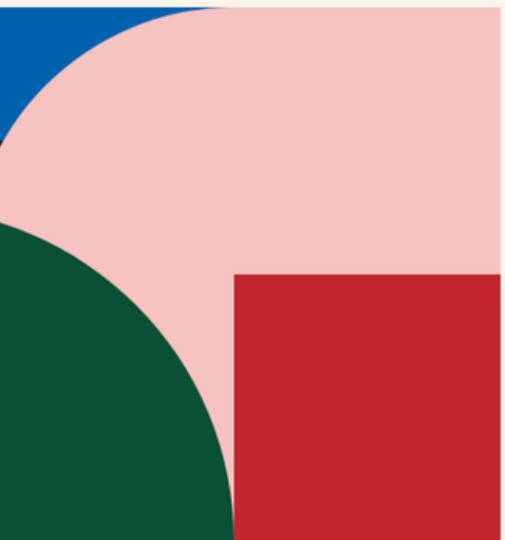
One problem that the last few solutions have is that we don't know the first value when we are at the second value and so we have to encode it initially in the group and then expand that encoding when we are at the second value. For full points this is rather hard and one way to achieve this is the following:

$$v_i = 0 \rightarrow p_{5h+2}, p_{5h+3}, p_{5h+4}, p_{5h+5}, p_{5h+6} = 00000$$

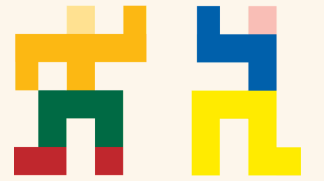
$$v_i = 1 \rightarrow p_{5h+2}, p_{5h+3}, p_{5h+4}, p_{5h+5}, p_{5h+6} = 00001$$

$$v_i = 2 \rightarrow p_{5h+2}, p_{5h+3}, p_{5h+4}, p_{5h+5}, p_{5h+6} = 00010$$

$$v_i = 3 \rightarrow p_{5h+2}, p_{5h+3}, p_{5h+4}, p_{5h+5}, p_{5h+6} = 00101$$



ELEVATOR



Subtask 4 - K = 25

Then for two values:

$v_i, v_j = 0? \rightarrow p_{5h+2}, p_{5h+3}, p_{5h+4}, p_{5h+5}, p_{5h+6} = 01110, 11010, 11100, 11110$

$v_i, v_j = 1? \rightarrow p_{5h+2}, p_{5h+3}, p_{5h+4}, p_{5h+5}, p_{5h+6} = 11001, 11011, 11101, 11111$

$v_i, v_j = 2? \rightarrow p_{5h+2}, p_{5h+3}, p_{5h+4}, p_{5h+5}, p_{5h+6} = 01011, 10011, 10110, 10111$

$v_i, v_j = 3? \rightarrow p_{5h+2}, p_{5h+3}, p_{5h+4}, p_{5h+5}, p_{5h+6} = 00111, 01101, 01111, 10101$

