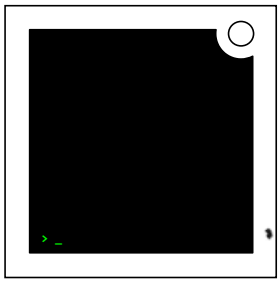
On the Subject of Sysadmin



*sudo rm -rf /**

The module is a console. When it is waiting for command input you will see green > _ symbols.

This console is connected to a bus topology network. The network contains 100 nodes. Each node is numbered from 0 to 99 without repetitions. All nodes are sorted by their id. Up to ten of these nodes are servers, the rest are data storages. Each storage is 1 TB in size. There is a virus on the network that damages the nodes. You can fix it, but you need to do it as soon as possible - the longer you are idle, the more nodes will be damaged.

If the node is damaged, then find the error code in the table on page #3. Then calculate the value in the last column on the same row (using the table on the current page). If this value is greater than 9 or less than 0, then subtract or add 10 until the number is in the range [0-9]. In the same column as the error code, find all codes that contains the calculated value. The recovering code is equal to the first characters of the found codes, starting from the next one after the error code and further down. This action is looped vertically, so after going through the last row, go back to the first.

Information on how to solve the module and in what cases the strikes will be received can be found on page #2. Page #2 contains a list and description of the commands available for entering into the console.

All commands (including codes) are case insensitive.

T	Starting time in minutes
F	Recovered nodes count
B	Batteries count
I	Indicators count
K	Max least significant digit of two factor codes. If there are no two factor codes, use solved modules count
R	Remaining minutes count
M	Modules count
S	Min digit in serial number
E	Strikes count

Commands:

- `serverlist` - Displays information about all servers. For each server, information about the id of the node and the required storage size for work will be provided. If storage nodes have already been allocated to this server, then the range of allocated nodes will also be displayed.
- `status` - Displays general information about the current state of the network. Here you can see the number of servers to which the storage nodes have been allocated; the required number of allocation that need to be made for the module to be solved; number of damaged and recovered nodes.
- `allocate {NODE_ID} {DIRECTION}` - Allocates the required number of nodes in the specified DIRECTION to the server with the specified NODE_ID. Direction must be one of the following: `left`, `down` (decreasing id), `right` or `up` (increasing id). In order for the command to work correctly, the following conditions must be met:
  - The selected node must be a server;
  - * The allocation nodes must be data storages (not servers);
  - There should be no attempts to allocate nodes with id less than 0 or greater than 99;
  - The server node, as well as the allocation nodes, should not be damaged;
  - * The allocation nodes should not be allocated to another server.

  If the conditions marked with an asterisk are not met, all allocations will be reverted and the strike will be counted.
- `debug {NODE_ID}` - Checks the status of a node by its NODE_ID. If it is damaged, then an error code will be displayed in the result.
- `recover {NODE_ID} {FIX_CODE}` - Attempts to recover data on a node by its NODE_ID using FIX_CODE. If FIX_CODE is correct, then the node will be recovered. If FIX_CODE is incorrect or this node is not damaged, then all allocations will be reverted and a strike will be counted.
- `commit` - Applies all changes. If the required number of allocations is reached, the module will be solved, otherwise all allocations will be reverted and a strike will be counted.
- `revert` - Reverts all allocations and gives a strike.
- `clear` - Clears the console

8BP5	52SA	G0ST	6XTT	7Y79	0JK0	FT37	965B	Z7C4	FZY3	I
6ZP8	6446	5L1O	4D9A	7T7X	8IWL	5841	56OW	4AX8	3WY0	S
3EAM	09R0	KAY6	2Z3O	7004	012J	1M67	2G6X	006K	437Y	M
523G	8MR3	JW99	6X6C	4HZZ	UMP9	5UCF	6S82	5IM7	7FX2	3 + K
V0SR	2TF5	4958	CYB6	2484	73T1	88K2	4Q1F	4L73	93HV	F + 5
0P3T	3X80	3577	5JVK	3W31	20ZK	1MD1	3QB2	5T5U	6PHO	K
3O4X	6N8N	46UC	5K27	0F5V	74GE	15UT	680W	CMK9	6WMJ	T
3D40	K329	1J7R	V5Z9	8LO4	5U9H	975Z	8S8C	J27H	8868	7 + E
2E3D	S5Y5	7Q12	0CU2	Y9DD	E4DR	6XXS	6FVU	547C	5TVT	E
6D44	742X	D86Q	S912	7B2Z	OEX9	BB35	7Y11	274A	836C	S
8567	PT68	5RN4	WIQ2	0KAY	0PQ8	6AG0	M7S4	2ROV	6RO0	T
391M	M4NF	ZUY2	67W9	70CW	6530	7F4Q	5PA5	693M	6W2F	M
57T1	14V2	0JW4	8TH7	WAX1	6SH6	7XGD	250K	78EU	4YKZ	B
86V2	1X2X	8NWD	478A	823O	L509	5BI0	2267	TL08	42J0	M + 2
4EUE	4355	TGZ2	4Y7Z	4769	5NCR	8A0V	A9E1	212K	1366	E
1RM6	6X6O	1PCO	7OXG	X8IZ	UHW4	G61J	01OA	083J	2WP1	R
9JZ8	4768	2KRB	5QNO	Y2GA	7A4T	GBZ2	XKU3	U291	65HQ	T + R
89DE	J9FZ	3ZJA	9MLG	C7FG	1ZWL	81J5	34LZ	W73I	SZG1	B + 2
130Q	886C	75ZC	8IK5	2C91	P84G	7055	30C2	FR8H	8A9S	4 + M
54JH	0168	128J	11JZ	22HW	16UD	2143	43CU	7N47	7S74	F
3G15	0Z4Z	6K8I	Y50L	961B	1576	0L60	IZ47	9K77	049I	K
51P0	37V1	HR71	5J46	V1VP	5340	1ZHJ	1KRZ	VF8H	241X	8 - S
7QU3	73ND	4998	74FU	E308	8V94	6UMZ	YBZ6	2Y99	103Q	K - F
0K4U	8EN8	89N2	0TZ5	6648	923B	8613	8F28	1KV4	1OD3	B
6ZG0	0978	41C5	8065	1D21	23JJ	57N2	130U	30ZB	P57Y	S
II6C	9139	51YS	F65N	0346	9SC5	25BI	I38I	632R	8NWB	F
924U	22S8	44HK	608D	7353	24J3	CW49	15QF	JV7M	2B7G	T
9988	UJ98	44B7	ZK58	0C7A	8Y69	9MPW	500G	634Y	DY5O	F + 7
B2J7	55WZ	GD9L	7PT6	H89I	7J15	3516	YBL4	664P	9D50	2 + B
L149	629Q	0ZQD	8SV1	N14K	243X	956O	12O0	7MZN	1054	B - T