Problem I. Polyomino Packing

Input file:	*standard input*
Output file:	*standard output*
Time limit:	5 seconds
Memory limit:	256 mebibytes

Little Dasha got a new game as a present. The box containing the game is lined into $h \times w$ squares of the same size. The box contains polyomino pieces. Each polyomino fully occupies from three to five squares, and the occupied area is side-connected. The polyominoes can be freely taken from the box one by one, both their surfaces are colored, so they can be rotated and flipped during the game. Additionally, there is a folded playing field in the box, and it has the size of $2h \times 2w$ squares.

Dasha laid the playing field on the floor and placed all the polyominoes on the field so that each polyomino fully occupied several squares, none sticked out of the field, and no two polyominoes had a common square.

The time came to put the game in the box. Dasha started to place the polyominoes, but it turned out this was not easy: she succeeded in placing some of them, but the shape of the remaining polyominoes did not allow to place them without overlapping the polyominoes already placed. Dasha returned all polyominoes on the field and sat nearby, thinking.

Solve a generalized version of Dasha's problem. Given the configuration of the polyominoes on the playing field, output how to place all of them in the box so that each polyomino fully occupies several squares, none stick out of the box, and no two polyominoes have a common square.

Input

The first line of the input contains two integers h and w: the height and width of the *box* ($20 \le h, w \le 100$). After that, the field containing the contents of the box is given: each of the following $2h$ lines contains exactly $2w$ characters. If a character is "." (dot, ASCII code 46), the corresponding square is empty. Otherwise, the character is one of the capital English letters ("A"–"Z"). Each polyomino is a region containing the same letter which consists of 3, 4 or 5 squares and is side-connected.

Output

Print exactly h lines, each containing exactly w characters: the way of placing the polyominoes in the box. Use the same format as the given field, with one obvious difference: there can be no empty squares left.

Each polyomino can be rotated and flipped. The letter corresponding to each polyomino can be changed arbitrarily. However, all polyominoes from the input must be placed in the box.

Note

Each test for this problem is generated by the following pseudorandom algorithm. First, the jury picks h and w, and then initializes a pseudorandom number generator by a randomly selected value.

After that, the generator is used to generate a sequence of polyomino types: a new type is selected uniformly at random from all the possible types as long as the total area of all polyominoes is less than $h \times w$. If the total area happened to become greater than $h \times w$, the whole sequence is generated again from the start. Note that there are two types of polyominoes consisting of three squares, five types consisting of four squares and 12 types consisting of five squares.

Finally, the polyominoes from the sequence are put on the field one by one. For each polyomino, the orientation, which consists of rotation and flipping, is selected randomly. Then, a position for the polyomino is selected uniformly at random from all positions possible at the moment.

In total, there are 81 tests in this problem. It is guaranteed that the answer exists in all the resulting tests.

In this problem, **half of the tests are open**. You can download all tests with odd numbers, including the example, via the link from the contest system.

Example

standard input	standard output
20 20 ....C.........LL..........T.BBBBB...H..... ..C...PPP..LL..S.......TT.........H..... ..C...P.P.....SSS...X.TT.........HHH.... ..CBBB......I......XX.........RRRR...... ..CB.......II...X...X.......FF...AA..... ...M......II....XXX.......M.FF..AAA..... ..MMM........S...X.YYY...MM.F........... ...M....M....S....YY...CC.M............. ..GGGG.MMM...S.........CC............... ..G.....M...GGG.FFFFF.......KK...K...... ...........II............C.KKGG..KK..... ...........II............C...G...K...... .................MM......CC.......K...... .KKK.............M......C..EEE.......... ..K..............MM.........E........... W..EEZZZZ...................E........... W...EZ..P...............RRR.H.....EEE... WFFFFF..P...............RAAAH.N....E.... W....G.PPP...........O..R..AH.NNN..E.... W...GG..............OO......H...N....MM. .....GG.............O......HS.....R.M.. ..UJJJJJ............O.........SCCC..R.... ..U............I....O..XXX...S.C...RR... ..UU.QQ.P.....FIIIOOO.........S.C...R.... .....Q..PPPM..F............CCS..P....... ...I....P.MM..FLLL.........CC.PPP....... ...I......MM..F....OOO..........P....... ..III.....A...F..X.O.....N.............. ..UUU.....AAA..XXX.......N......LLLLL... ....UU....A......X.......N.............. ...TTLLLLL.........VVA....N..........LLLL .TTT...E.........VV.A..PP..............L .......E.........P..A..PP....AAAAA....G. .....D.E..P.....PP.....B.U............G. ....DDDE..P....PPU....BB.UU...........G. .......E.PPPE..Z.U....BJUU...........GG. ....Y.......EZZZ.U.H...JJJ......LLLL.... ..YYY.....EEE.Z..U.HDDDJ.........L....... ..Y................HHHDD................	PPLLLNFFFFPPLLLPPPPP PPLLNNNJFPPLLJJJJJNN PJJJJNJJJPFFFFFNNNJN NNNJPFPPPPNNNLLLHNJJN LNPPPFFFJJNLLHHHNLJN LNJJPFJJJPNJJJJJHLLL LJJHHHNPPPLLLPPPHHHL LJHHNNNLPNNLNNPLLHNN LPPPFFFLLNNLNNLLPPPN HLLLPJJJLJJJJJHHHHPL HHHLPPPJPPFFNNFFFFFL HPNLPHNJPPLFNPPDDDDL JPNNNHNNNLLLHHPNHHHH JPPPNHHHPDPPHHJNNNNP JJLFFFFFPDDPPLJJJJPP PPLLLLJPPPDLLLBBBBBP NPPPHJJJNNNNNLPPPFFN NNNHHHDJLPFFFFFPPFPN PNPNHDDDLPPNNNNNFFPN PPPNNNNLLLPPLLLLLPPP