**ACSL Rack-O**

**CLASS / SOURCE NAME**

**Name your class acslint.**

**PROBLEM:**The game of Rack-O is played with each player being dealt 10 cards with numbers from 1 to 60 on them. Each player places those cards in the order dealt, from front to back, in a *rack*with slots for holding the cards. Through a series of draws from the draw *pile*, each player may replace a card in the rack with the drawn card. The goal is to have the cards in the rack in ascending order. The first player to do so is the winner.

In the ACSL version, you will be given the number of slots, *s*, and the number of cards, *n*, in the game. Each card has a unique number, between 1 and *n*, inclusive. You will be given *s*cards to place in the rack, front to back. You will then be given a set of cards forming the draw pile, that you will try to play in that order. If the cards in the rack are not already in ascending order, for each card that is drawn, follow these rules in this order:

1. Except for the first card in the rack, if the drawn card is 1 less than any card already in the rack, replace the card in the slot before that card with the drawn card.
2. Except for the last card in the rack, if the drawn card is 1 greater than any card already in the rack, replace the card in the slot after that card with the drawn card.
3. For any three cards that are in adjacent slots in the rack, if the card in the middle is not between the first and last card, but the drawn card is between the first and last card, replace the one in the middle with the drawn card.
4. If the drawn card is less than the second card in the rack and the first card in the rack is greater than that second card, replace that first card in the rack with the drawn card.
5. If the drawn card is greater than the next to last card in the rack and the last card in the rack is less than that next to last card, replace that last card with the drawn card.
6. Otherwise, do not use that card and go to the next one.

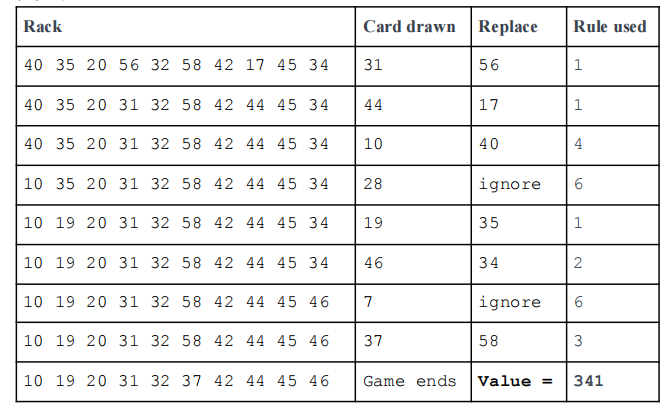
As soon as the cards in the rack are in ascending order, the game ends and no more cards are drawn. The value of the rack is the sum of its cards plus bonus points for having a 3 or more card sequence. Add 5 points per number for each card that is in a sequence. For example, 45, 46, 47 is 15 points and 3, 4, 5, 6 is 20 points. Otherwise, when you have finished looking at every card in the draw pile, the value of the rack is the opposite of how many *step-downs*there are. For example, in the list 40, 47, 20, 56, 15, 16, 17, there is a step-down after 47 and another one after 56. The sequence of 15, 16, 17 is not counted because the entire rack is not in ascending order. Therefore, the value of that rack is -2.

EXAMPLE

|  |  |
| --- | --- |
| **Input** | **Output** |
| 10 60  40 35 20 56 32 58 42 17 45 34  31 44 10 28 19 46 7 37 16 2 | 341 |

Explanation:

See the table below for the sequence of moves. The final rack is in ascending order so output the sum of the numbers in the rack (326) plus the bonus for a 3-card sequence (15) for the total value of 341.



**INPUT:**There will be three strings: the number of slots and the number of cards, the cards in the rack, and the cards in the draw pile. All numbers will be separated by a single space.

**OUTPUT:**Output the value of the player’s rack as described above.

|  |  |
| --- | --- |
| **SAMPLE INPUT** | **SAMPLE OUTPUT** |
| 10 60  40 35 20 56 32 58 42 17 45 34  31 44 10 28 19 46 7 37 16 2 | 341 |
| 15 90  15 12 18 9 28 17 46 51 7 53 65 70 74 84 47  45 73 3 52 54 16 21 44 87 40 68 30 20 | 752 |
| 12 130  20 110 30 16 84 40 91 69 75 7 81 15  39 47 114 55 35 71 25 123 51 23 34 10 77 36 115 | 656 |
| 8 100  6 13 47 62 32 70 76 12  3 67 80 10 39 44 2 43 40 85 21 33 4 52 | 421 |
| 12 110  44 35 22 25 79 100 85 69 87 3 56 28  97 10 48 43 42 21 81 47 86 88 80 54 24 50 | -1 |

-----------------------------------------------------------------------------------------以下为中文翻译，仅供参考-----------------------------------------------------------------------------------------------------------

**问题：**Rack-O 游戏中，每位玩家都会收到 10 张卡牌，每张卡牌上印有一个位于 1 到 60 之间的数字。每位玩家按照发牌顺序，将这 10 张卡牌从前至后摆放在一个带卡槽的 *架子*上。每位玩家从抽牌 *堆*中抽牌，在一系列抽取的过程中，玩家可以用抽到的卡牌替换在牌架上的一张卡牌。游戏目标是使得牌架上所有卡牌按升序排列。第一位达成游戏目标的玩家获胜。

在 ACSL 版的 Rack-O 游戏中，玩家会被告知卡槽数量 *s*，以及卡牌数量 *n*。每张卡牌上都有一个在 1 到 *n*之间（包括 1 和 *n*）的专属数字。玩家将会收到*s* 张卡牌，然后将这些卡牌从前至后摆放在牌架上。除此之外，玩家还会收到一叠卡牌，形成一个抽牌堆，按照卡牌叠放顺序依次抽牌。如果牌架上的卡牌未按升序排列，那么对于每一张抽到的卡牌都要按以下顺序遵循下列规则：

1. 除牌架上第一张卡牌之外，如果抽到的卡牌的数字比在牌架上某一张卡牌上的数字小 1，则用这张抽到的卡牌替换牌架上这张牌前面的卡牌。

2. 除牌架上最后一张卡牌之外，如果抽到的卡牌的数字比牌架上某一张卡牌上的数字大 1，则用这张抽到的牌替换牌架上这张牌后面的卡牌。

3. 对于牌架上任意三张在相邻卡槽中的卡牌，如果在中间位置的卡牌上的数字并不介于这三张中第一张卡牌和最后一张卡牌的数字之间，但抽到的卡牌上的数字介于其间，则用抽到的卡牌替换在中间位置的卡牌。

4. 如果抽到的卡牌比牌架上第二张卡牌小且牌架上第一张卡牌比第二张卡牌大，则用抽到的卡牌替换牌架上第一张卡牌。

5. 如果抽到的卡牌比牌架上倒数第二张卡牌大且牌架上最后一张卡牌比倒数第二张卡牌小，则用抽到的卡牌替换最后一张卡牌。

6. 否则，放弃使用抽到的卡牌，抽取下一张。

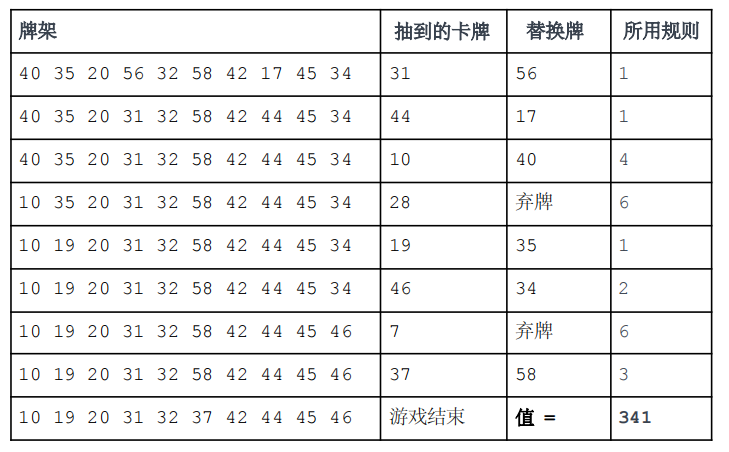
一旦牌架上的牌按升序排列，游戏结束，不再抽牌。牌架值等于牌架上所有卡牌之和再加上奖励分。如果牌架上出现 3 张或更多张数字连续的卡牌，那么这些数字连续的卡牌每张都可以再加上5 分的奖励分。例如，如果牌架上出现数字 45、46、47，就可以额外加上 15 分的奖励分；如果牌架上出现数字 3、4、5、6，就可以额外加上 20 分的奖励分。但是，如果抽牌堆中的卡牌全部抽取完毕，但牌架上的所有卡牌未按升序排列，那么牌架值将会等于牌架上卡牌数字 *递减*次数的相反数。举例来说，如果牌架上的卡牌数字序列为 40、47、20、56、15、16、17，那么数字 47 之后紧接着是较小的数字 20，数字 56 之后紧接着是较小的数字 15，这两次递减都会计入次数。并且，由于牌架上所有卡牌未按升序排列，因此 15, 16, 17 不计奖励分。所以最后牌架值等于 -2 。

样例：

|  |  |
| --- | --- |
| **输入** | **输出** |
| 10 60  40 35 20 56 32 58 42 17 45 34  31 44 10 28 19 46 7 37 16 2 | 341 |

解析：

操作顺序详见下表。最终牌架上所有卡牌按照升序排列，因此输出牌架上所有卡牌之和（326）再加上其中 3 张数字连续卡牌的奖励分（15），得到最终牌架值 341。



**输入**：输入三个字符串：卡槽数量及卡牌数量；牌架上的卡牌；抽牌堆中的卡牌。各个数字之间用一个空格隔开。

**输出**：如上所述，输出玩家的牌架值。

|  |  |
| --- | --- |
| **样例输入** | **样例输出** |
| 10 60  40 35 20 56 32 58 42 17 45 34  31 44 10 28 19 46 7 37 16 2 | 341 |
| 15 90  15 12 18 9 28 17 46 51 7 53 65 70 74 84 47  45 73 3 52 54 16 21 44 87 40 68 30 20 | 752 |
| 12 130  20 110 30 16 84 40 91 69 75 7 81 15  39 47 114 55 35 71 25 123 51 23 34 10 77 36 115 | 656 |
| 8 100  6 13 47 62 32 70 76 12  3 67 80 10 39 44 2 43 40 85 21 33 4 52 | 421 |
| 12 110  44 35 22 25 79 100 85 69 87 3 56 28  97 10 48 43 42 21 81 47 86 88 80 54 24 50 | -1 |

注意：

(1) 样本数据仅为部分测试数据，测试用例全部通过不代表通过本题。

(2) 你必须通过数据库中所有的测试点才能获得该题满分。

(3) java 语言里面的 class name（类名）需要用本题的 Source file name（即：acslint）。

(4) 平台判分规则为调取 5 次 input 值，每次单独判分。