

@Nabetani 算数数学問題事業部 鍋谷武典さんの問題に挑戦中！

最寄りの素数

最寄りの素数

【概要】

右図のように、1以上のすべての整数をならべます。
入力で指定された数に最も近い素数を求めてください。

【入力と出力】

入力は

14

のような形で、標準入力から来ます。
普通に10進数です。素数が来ることはありません。

出力は、入力の最寄りの素数です。
最寄りの素数が複数ある場合は、小さい順にコンマ区切りで出力してください。

1	2	5	10	17	26	37	50	65	82	101	122
4	3	6	11	18	27	38	51	66	83	102	123
9	8	7	12	19	28	39	52	67	84	103	124
16	15	14	13	20	29	40	53	68	85	104	125
25	24	23	22	21	30	41	54	69	86	105	126
36	35	34	33	32	31	42	55	70	87	106	127
49	48	47	46	45	44	43	56	71	88	107	128
64	63	62	61	60	59	58	57	72	89	108	129
81	80	79	78	77	76	75	74	73	90	109	130
100	99	98	97	96	95	94	93	92	91	110	131
121	120	119	118	117	116	115	114	113	112	111	132
144	143	142	141	140	139	138	137	136	135	134	133

【入出力の例】

入力

出力

周囲の状況

14

7, 13, 23

1	2	5	10	17	26	37
4	3	6	11	18	27	38
9	8	7	12	19	28	39
16	15	14	13	20	29	40
25	24	23	22	21	30	41
36	35	34	33	32	31	42
49	48	47	46	45	44	43

12

7, 11, 13, 19

1	2	5	10	17	26	37
4	3	6	11	18	27	38
9	8	7	12	19	28	39
16	15	14	13	20	29	40
25	24	23	22	21	30	41
36	35	34	33	32	31	42
49	48	47	46	45	44	43

22

13, 23

4	3	6	11	18	27	38
9	8	7	12	19	28	39
16	15	14	13	20	29	40
25	24	23	22	21	30	41
36	35	34	33	32	31	42
49	48	47	46	45	44	43
64	63	62	61	60	59	58

32

31

8	7	12	19	28	39	52
15	14	13	20	29	40	53
24	23	22	21	30	41	54
35	34	33	32	31	42	55
48	47	46	45	44	43	56
63	62	61	60	59	58	57
80	79	78	77	76	75	74

		47	46	45	44	43	56	71
		62	61	60	59	58	57	72
		79	78	77	76	75	74	73
95	59, 97, 139	98	97	96	95	94	93	92
		119	118	117	116	115	114	113
		142	141	140	139	138	137	136
		167	166	165	164	163	162	161

		74	73	90	109	130	153	178
		93	92	91	110	131	154	179
		114	113	112	111	132	155	180
134	113, 131, 157	137	136	135	134	133	156	181
		162	161	160	159	158	157	182
		189	188	187	186	185	184	183
		218	217	216	215	214	213	212

【補足】

入力の範囲は、1以上、百万以下です。入力の値は素数ではありません。

出力には、百万より大きな数が含まれているかもしれません。

距離は、マスの中心を結ぶ線分の長さです。マス目は正方形とします。

不正な入力（“百万”、“1+2+3”など）に対応する必要はありません。

ライブラリなどに素数の判定や列挙を行う関数などがある場合、遠慮なく使っていて構いません。

【解答方法】

■挑戦言語は下記のプログラム言語選択で選択可能なものであれば何でもOKです。

1. 自分の書いたプログラム言語を選択
2. 解答欄にソースコードを記入
3. 送信前に「提出前に確認」ボタンをクリック（構文エラーがないかどうかチェックできます）
4. 「解答コードは正常に実行されました」というメッセージを確認の上、「解答を送信」ボタンで解答してください。

■この問題にはテストケースが13件用意されています。すべてに通れば正解です！

【採点について】

採点は「ideone」を使ってプログラムを実行し、標準入力および標準出力のテストケースと照合して正誤を判定します

各言語の標準入力と標準出力は[こちら](#)を参考にしてください

※なおCodeIQで使用しているideoneは企業版のため、webで公開されている[コンシューマー版ideone](#)とは対応言語・バージョン・挙動が異なる場合があります。

企業版ideoneの対応バージョンは、「提出前チェック」の結果とともに表示されます。



```
# coding: UTF-8

import math

# 素数判定
def is_prime(q):
    q = abs(q)
    if q == 2: return True
    if q < 2 or q%2 == 0: return False
    return pow(2, q-1, q) == 1

# マップ内X座標取得
def get_MapX(v):
    x1 = math.ceil(math.sqrt(v))
    if x1 - 1 <= (x1*x1) - v:
        return x1
    else:
        return int(math.fabs((x1*x1) - v + 1))

# マップ内Y座標取得
def get_MapY(v):
    y1 = math.ceil(math.sqrt(v))
    if (y1*y1) - v <= y1 - 1:
        return y1
    else:
        return int(math.fabs((y1-1)*(y1-1) - v))

if __name__ == "__main__":

    try:
        while True:
            # 標準入力受け付け
```



公式Facebookページ

公式Twitterアカウント

CodeIQ

CodeIQ MAGAZINE

CodeIQ JOBS

Read CodeIQ on RSS

CodeIQとは

CodeIQ（コードアイキュー）とは、自分の実力を知りたいITエンジニア向けの、実務スキル評価サービスです。

[プレスリリース](#)

[CodeIQ](#)

[CodeIQ Magazine](#)

[CodeIQ JOBS](#)

CodeIQご利用にあたって

- [利用規約](#)
- [プライバシーポリシー](#)
- [お問い合わせ](#)
- [エンジニア採用ご担当者様へ](#)

関連サイト

- [ATND](#)
- [CODE VS](#)
- [サンカク](#)
- [カクシン](#)
- [リクナビNEXT](#)
- [リクナビNEXT ITキャリア](#)
- [リクナビNEXT Tech総研](#)