ch.	v _0-	0.2	~	Logo	0.5	'5°	٥٧	a.º
00653	SAR SACA	CDO 23 PR	ST	UDENT RE	PORT	Seb Straff	3CD065385	**************************************
	DETAILS Name	360065 3BR13 CD065 3BR	38kg 3ch 0 ch	3C10653BR?3C10653B	PORT BODG	38R23CID 6538FF	3500° 3500653E	,0065 3 CO065 3 8 C
13cD0	PAVAN KUMAR Roll Number	R M JANGANI						
	3BR23CD065		aRiv					, 5
20	EXPERIMENT  Title  SPECIAL STRING  Description	3CD0653BR73CD0653BR	23CD0653BR23CD065	3600653HR13C006	236H23CD0653HR23CD6	ARA SCHOOL SHAN	6434RL3CDO6538	,00653BR/3C006
3BR1	Description	(3BR)3CD	2005 3BRIV	300065	23°CV 200653	273cDC	E SPENSO	30063365
23CD06	Alice has a stri	ing A consisting of lits characters with t	owercase English	letters. Her friend	gives her another	string S and ask	s her to modify st	ring
2	But, to achieve	the above task, Alio	ce must follow the	e below steps:				SBRU
08	1. Choose a ch	aracter from string	S that has the mi	nimum ASCII dista	nce from the ith	character in string	g A	
0065 3S	Replace the ith	character in string						JA to 223choś
	the characters	find and return an in string S. Return (		•			ed to modify string	JA to
3BR23	Sample Input:							<u>~</u>
30	abcd							,0065385
0	xyz							
13cDo	Sample Output	t:						SARRIAC
								38PC
,0653R	Source Code: 6 3 CO	3C00653BR23C00653BR23C5	13C0065381	38k23C106538k23C106538k	3BR13CDO653BR13CDO653	PP SELLINGER SELLINGER	COOKS STATE OF THE PARTY OF THE	A CHARLES REPORT
3BK13	38E13C106538	9062 3BK 3C106	Job's 3HR130	3C1065345	23c1065	8823°C10	So S	A Proposition of the series
	38EV3CV	38F <sup>23</sup> CD(65 <sup>3</sup> 3B) <sup>C</sup>	38k1/3CD0623	653HR13C	0065 34F	SCIO CO.	September Septem	and the state of t
		38RP3CV	13C00653BR	381 <sup>3</sup> 51065	38R2 3CV	Popular,	6.00 60 5 5 5 60 60 60 60 60 60 60 60 60 60 60 60 60	A Profits
			3	30	60	921	233	16A

```
def min_ascii_distance(A, S):
        total_distance = 0
        found_all = True
        for char_a in A:
            \mbox{\#} Find the minimum ASCII distance character in S
            min_distance = float('inf')
            for char_s in S:
                distance = abs(ord(char_a) - ord(char_s))
                if distance < min_distance:</pre>
                    min_distance = distance
            # If the character from A is not in S, we add the minimum distance
            if min_distance != 0:
                found_all = False
                total_distance += min_distance
        return total_distance if not found_all else 0
   # Sample Input
   A = str(input())
   S = str(input())
   # Finding the minimum total ASCII distance
   result = min_ascii_distance(A, S)
   print(result) # Output: 86
RESULT
  5 / 5 Test Cases Passed | 100 %
   BRI
```