=== Compiler online time ===

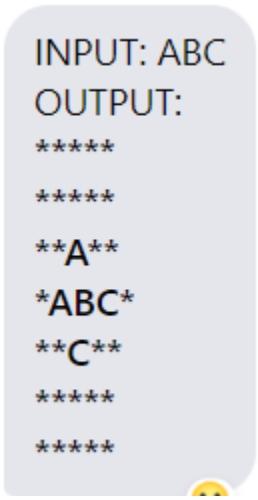
A1: 30+10
A2: 25
B1: 30+5
B2: 30

=== Online 1 (Basics) ===

A1 (Input will be A/B/C/..../H, time 40 mins)

Input => Output Format:
A => 17
B => 16
.
.
.
G => 11
H => 10

A2 (Input always 3 characters, time 20 mins)

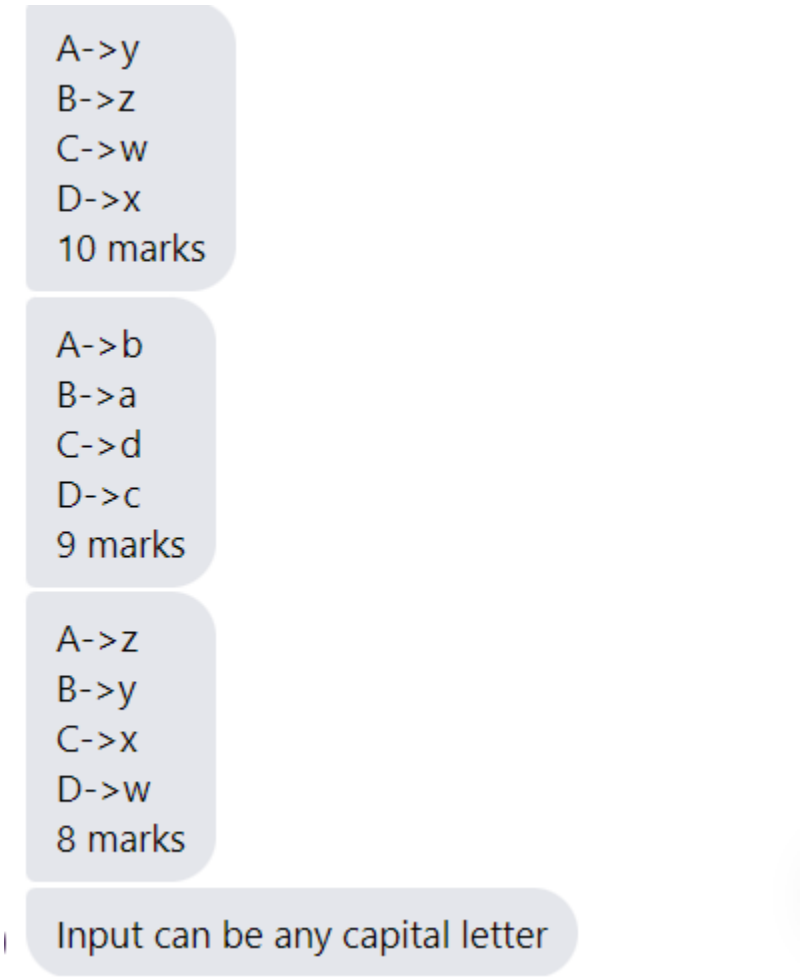


B1 (Input will be a to m always, time 30 mins)

Input => Output Format:
a => Z

b => X
c => V
.
.
.
m => B

B2



There are 3 questions. Solve according to the mark you want

=== Online 2 (If-else) ===

A1

3 sides (single digit) of a triangle will be given. Determine which family it belongs to (Equilateral/Isosceles/Scalene)

A2
Input: A number between 1-100. Multi-digit input consumption part was provided
Output: Determine the grade
Sample I/O:
80-100 => A+
70-79 => A
.
.
.
0-39 => F

B1
Input: Any character (Any means any. Lowercase/uppercase letters, other ascii characters, anything can be input. One teacher even pressed F10 as input lul)
Output: Determine whether the ascii character is vowel/consonant/other

B2
Input: Two numbers that depict the abscissa and ordinate of a point in the Cartesian coordinate system. To distinguish between pos & neg numbers, pos numbers will always start with '0' (e.g. 05, 07) & neg numbers will start with '-' (e.g. -5, -7). Please note that input will always be two characters
Output: Determine which quadrant/axis the point lies in

=== Online 3 (Loop, stack etc) ===

A1
Input: Any 3-9 base number
Output: Convert it to binary
Time: 45 mins

A2

একটা নাম্বার ইনপুট নাও, ১ থেকে ঐ নাম্বারের মাঝে কতগুলো ফল্টি নাম্বার আছে, বের কর।

ডেফিনেশনঃ ফল্টি নাম্বার হচ্ছে যাদের প্রকৃত উৎপাদকের যোগফল, ঐ নাম্বার অপেক্ষা বেশি

১২ একটা ফল্টি নাম্বার
কারণ, ১২ এর প্রকৃত উৎপাদক ২,৩,৪,৬ এর যোগফল, ১৫ > ১২

Sample I/O
In: 10
Out: 0
In: 20
Out: 3

সময়, ৩৫ মিনিট

B1

Selection sort (a picture of selection sort simulation was provided. Also, since we did offline with insertion sort, the modification became comparatively less painful. I had to change only the sorting algo part, the input consumption and array printing part were the exact same as insertion sort; time 30+10)

B2

Input: An integer, N
Output: Print all the co-primes of N from numbers 2 to N **&** the total number of co-primes (C code of GCD was provided)
TIme: 40 mins