

## Indicaciones

- Recuerda que la **tarea es individual**. Los casos de copia/plagio serán sancionados con nota cero (0) en la asignatura.
- La tarea será publicada en Canvas pero **se entrega por la plataforma Gradescope**, además debe considerar:
  - Se le proporcionará un archivo `solution.py`. Usted deberá editarlo en las secciones correspondientes.
  - Al finalizar, solo subir el archivo `solution.py` (**NO cambiar el nombre del archivo y NO comprimirlo**).
  - Cada pregunta tiene casos de prueba, evalúe su solución con cada caso.
  - **La respuesta debe ser impresa con `print()`**.
  - Al momento de la impresión de la respuesta, no adicionar texto. Imprimir únicamente el resultado que pide el ejercicio.

## Problema 1 : Multiplicación de números - (5 pts)

Sofía tiene un examen de matemáticas pronto y necesita de tu ayuda para aprenderse las tablas de multiplicar. Escribe un programa que reciba como dato de entrada dos números **N** y **M**. Usted deberá imprimir la tabla de multiplicar del número **N** hasta **M**, tomando en cuenta los 10 primeros múltiplos. En caso de que **N** sea mayor que **M**, invertir los valores e ir de **M** hacia **N**.

La plantilla esta implementada con funciones, por lo que no es necesario leer los inputs, simplemente asumir que estos ya existen. De igual manera, para la impresión del output.

### Ejemplo 1

Input

```
1 N: 5
2 M: 13
```

## Output

```
1 5 10 15 20 25 30 35 40 45 50
2 6 12 18 24 30 36 42 48 54 60
3 7 14 21 28 35 42 49 56 63 70
4 8 16 24 32 40 48 56 64 72 80
5 9 18 27 36 45 54 63 72 81 90
6 10 20 30 40 50 60 70 80 90 100
7 11 22 33 44 55 66 77 88 99 110
8 12 24 36 48 60 72 84 96 108 120
9 13 26 39 52 65 78 91 104 117 130
```

## Ejemplo 2

### Input

```
1 N: 36
2 M: 39
```

### Output

```
1 36 72 108 144 180 216 252 288 324 360
2 37 74 111 148 185 222 259 296 333 370
3 38 76 114 152 190 228 266 304 342 380
4 39 78 117 156 195 234 273 312 351 390
```

## Ejemplo 3

### Input

```
1 N: 18
2 M: 9
```

### Output

```
1 9 18 27 36 45 54 63 72 81 90
2 10 20 30 40 50 60 70 80 90 100
3 11 22 33 44 55 66 77 88 99 110
4 12 24 36 48 60 72 84 96 108 120
5 13 26 39 52 65 78 91 104 117 130
6 14 28 42 56 70 84 98 112 126 140
7 15 30 45 60 75 90 105 120 135 150
8 16 32 48 64 80 96 112 128 144 160
```

```
9 17 34 51 68 85 102 119 136 153 170
10 18 36 54 72 90 108 126 144 162 180
```

## Problema 2: Validación de contraseñas (5 pts)

Su labor en este problema es verificar y validar un grupo de contraseñas separadas por el carácter "/" según los siguientes criterios:

- Que tenga una longitud mayor o igual a 8 caracteres.
- Que tenga mínimo una letra mayúscula y mínimo una letra minúscula.
- Que incluya mínimo uno de los siguientes caracteres especiales ("-", "\*", "?", "!", "@")
- Que NO tenga espacios en blanco

El programa deberá imprimir el número de contraseñas válidas, el número de contraseñas inválidas y la lista de contraseñas válidas.

TIP: Usar listas.

Para la lectura de los inputs siga el formato de los ejemplos. De igual manera, para la impresión del output.

### Ejemplo 1

#### Input

```
1 contraseñas: Ut3c111?/123456789/S3r3n4 D3 L30n@/
   NothingElseMatters1*!/LimaPeru1
```

#### Output

```
1 validas: 2
2 invalidas: 3
3 ['Ut3c111?', 'NothingElseMatters1*!']
```

### Ejemplo 2

#### Input

```
1 contraseñas: estaesunacontraseña/Comput3rCci3nc3/programacion1
   -2023
```

## Output

```
1 validas: 0
2 invalidas: 3
3 []
```

## Ejemplo 3

### Input

```
1 contraseñas: contraseña1/contraseña2/Contraseña3-/contraseña4/
               contraseña5
```

### Output

```
1 validas: 1
2 invalidas: 4
3 ['Contraseña3-']
```

## Problema 3: Encriptación de mensajes - (10 pts)

En criptografía, el cifrado César es una de las técnicas de cifrado más simples y usadas. Se trata de un cifrado por sustitución en el que una letra es reemplazada por otra letra que se encuentra un número fijo de posiciones más adelante en el alfabeto. Por ejemplo, con un desplazamiento de 3, la A sería sustituida por la D (situada 3 lugares a la derecha de la A).

Deberá escribir un programa que reciba un número **N** indicando el desplazamiento a la derecha, el mensaje a encriptar e imprimir el mensaje encriptado. Sin embargo, tu tarea no acaba ahí. Recibirás además un *string* **S** con las 26 letras del alfabeto en desorden. Deberás reemplazar los caracteres del primer mensaje encriptado con su equivalencia en el nuevo string del alfabeto en desorden. Por ejemplo, si **S** es igual a **gfheidjckblamznyox-pwqvrust**, deberás reemplazar las "a" del primer mensaje por "g", las "B" por "f" y así sucesivamente.

NOTA: Tratar mayúsculas como si fuera minúsculas, es decir, "H" es igual a "h".

Para la lectura de los inputs siga el formato de los ejemplos. De igual manera, para la impresión del output.

## Ejemplo 1

### Input

```
1 N: 3
2 mensaje: Hola mi nombre es Paolo Morey y estudio en UTEC
3 S: gfheidjckblamznyoxpwqvrust
```

### Output

```
1 mensaje encriptado: lxne ya oxyiqc cv pexnx yxqcf f cvrujax co
  urcd
```

## Ejemplo 2

### Input

```
1 N: 5
2 mensaje: Ciencia de la Computacion
3 S: zyxwvutrsqponmlkjihgfedcba
```

### Output

```
1 mensaje encriptado: rmqhrmu sq ju rgifaburmgh
```