Example 2 (Worksheet) — Shift Cipher with $k = 11$

Goal. Encrypt the message `STOP GLOBAL WARMING` using Caesar's shift cipher with $k = 11$.

Big idea (the "why"):

We model letters as numbers in $\mathbb{Z}_{26}$ so that a shift is just *modular addition*. This keeps us in the alphabet and gives the wrap-around from Z back to A.

$$\text{A} = 0,\ \text{B} = 1,\ \ldots,\ \text{Z} = 25 \qquad\qquad E_k(p) = (p + k) \bmod 26.$$

For this example, $k = 11$.

Step 1 — Normalize and map letters → numbers

We use uppercase and keep spaces. Convert each letter of `STOP GLOBAL WARMING` to its number:

$$\underbrace{\texttt{STOP}}_{18\ 19\ 14\ 15} \quad \underbrace{\texttt{GLOBAL}}_{6\ 11\ 14\ 1\ 0\ 11} \quad \underbrace{\texttt{WARMING}}_{22\ 0\ 17\ 12\ 8\ 13\ 6}.$$

Step 2 — Apply the shift $k = 11$ (add 11 mod 26)

Compute $c \equiv p + 11 \pmod{26}$ for each number. Do the wrap when you go past 25.

$$\begin{aligned} \texttt{STOP}: &\quad 18, 19, 14, 15 \mapsto 3, 4, 25, 0 \\ \texttt{GLOBAL}: &\quad 6, 11, 14, 1, 0, 11 \mapsto 17, 22, 25, 12, 11, 22 \\ \texttt{WARMING}: &\quad 22, 0, 17, 12, 8, 13, 6 \mapsto 7, 11, 2, 23, 19, 24, 17. \end{aligned}$$

Step 3 — Map numbers → letters and keep spaces

$$3, 4, 25, 0 \ \mid\ 17, 22, 25, 12, 11, 22 \ \mid\ 7, 11, 2, 23, 19, 24, 17 \quad \Rightarrow \quad \boxed{\texttt{DEZA RWZMLW HLCXTYR}}$$

Helpful tips & common pitfalls

- **A=0, not 1.** Off-by-one mistakes are the #1 bug.
- **Wrap cleanly:** if $p + k \geq 26$, subtract 26 (i.e., reduce mod 26).

- **Spaces/punctuation** pass through unchanged; only letters get shifted.
- **Decrypting** with $k = 11$ is the same as adding -11 (or $+15$) mod 26.

Practice (your turn!)

Problem A (easier). Encrypt with $k = 4$: `MATH IS FUN`

Why: smaller shift, shorter phrase—perfect confidence builder.

Problem B (similar). Decrypt with $k = 11$: `SPWWZ HZCWO`

Tip: subtract 11 mod 26 or add 15.

Problem C (harder). Unknown k. Decrypt the Caesar ciphertext: `P HT HA AOL WHYR`
Hints: a one-letter word is often `I` or `A`. The block `AOL` frequently shows up when "`THE`" is encrypted with $k = 7$.

Reflection. In one sentence: explain why modular arithmetic guarantees a valid letter after every shift.