Example
Let the check digit \( a_4 = (7a_1 + 2a_2 + 9a_3) \mod 10 \)
Will this catch all single-digit errors in the first digit?

Correct Code: \( a_1a_2a_3 \) (or \( a_1a_2a_3a_4 \))
Incorrect Code: \( e_1a_2a_3 \) (or \( e_1a_2a_3a_4 \))

So the correct check digit is \( (7a_1 + 2a_2 + 9a_3) \mod 10 \) and
the incorrect check digit is \( (7e_1 + 2a_2 + 9a_3) \mod 10 \)

The error will NOT be caught if
\[
(7a_1 + 2a_2 + 9a_3) - (7e_1 + 2a_2 + 9a_3) \text{ is a mult. of 10}
\]
\[
\Rightarrow 7a_1 - 7e_1 \text{ is a mult. of 10}
\]
\[
\Rightarrow 7(a_1 - e_1) \text{ is a mult. of 10}
\]
\[
\Rightarrow a_1 - e_1 \text{ is a mult. of } \frac{10}{7}
\]

Multiples of \( \frac{10}{7} \) are \( \{0, \frac{10}{7}, \frac{20}{7}, \frac{30}{7}, \frac{40}{7}, \frac{50}{7}, \frac{60}{7}, \frac{10}{7}, \frac{80}{7}, \ldots \} \)

If \( a_1 - e_1 = 0 \), then \( a_1 = e_1 \), so we did not really make an error.

If \( |a_1 - e_1| = \frac{10}{7} \), then \( a_1 \) and \( e_1 \) are digits separated by \( \frac{10}{7} \) units.
The diff. bt. digits is always an integer, so this and all other non-integer mult. of \( \frac{10}{7} \) are impossible.

If \( |a_1 - e_1| = 10 \), then \( a_1 \) and \( e_1 \) are digits separated by 10 units.
Digits are never separated by more than 9 units,
so this and all higher mult. of \( \frac{10}{7} \) are impossible.

\( \therefore \) all single-digit errors in the first digit will be caught.
Example

Let the check digit \( a_4 = (7a_1 + 2a_2 + 9a_3) \mod 10 \)

Will this catch all single-digit errors in the second digit?

Correct code: \( a_1a_2a_3 \) (or \( a_1a_2a_3a_4 \))

Incorrect code: \( a_1e_2a_3 \) (or \( a_1e_2a_3a_4 \))

So the correct check digit is \( (7a_1 + 2a_2 + 9a_3) \mod 10 \) and the incorrect check digit is \( (7a_1 + 2e_2 + 9a_3) \mod 10 \).

The error will \textbf{not} be caught if

\[
(7a_1 + 2a_2 + 9a_3) - (7a_1 + 2e_2 + 9a_3) \text{ is a mult. of } 10
\]

\[\Rightarrow 2a_2 - 2e_2 \text{ is a mult. of } 10
\]

\[\Rightarrow 2(a_2 - e_2) \text{ is a mult. of } 10
\]

\[\Rightarrow a_2 - e_2 \text{ is a mult. of } \frac{10}{2} = 5.
\]

Multiples of 5 are \( \pm 5, 10, 15, 20, ... \)

If \( a_2 - e_2 = 0 \), then \( a_2 = e_2 \) so we did not really make an error.

If \( |a_2 - e_2| = 5 \), then \( a_2 \) and \( e_2 \) are digits sep. by 5 units.

These pairs are \( 0+5, 1+6, 2+7, 3+8, \) and \( 4+9 \).

These errors will \textbf{not} be caught.

If \( |a_2 - e_2| = 10 \), then \( a_2 \) and \( e_2 \) are digits sep. by 10 units.

Digits cannot be separated by more than 9 units, so this and all higher mult. of 5 are impossible.

:: All single-digit errors in the second digit

\underline{EXCEPT} those sep. by 5 units (0+5, 1+6, 2+7, 3+8, and 4+9)

will be caught.
Example
Let's keep the check digit $a_4 = (7a_1 + 2a_2 + 9a_3) \mod 10$

Will this catch all transposition errors in the **first two digits**?

**Correct code:** $a_1a_2a_3$ (or $a_1a_2a_3a_4$)

**Incorrect code:** $a_2a_1a_3$ (or $a_2a_1a_3a_4$)

So the correct check digit is $(7a_1 + 2a_2 + 9a_3) \mod 10$ and
the incorrect check digit is $(7a_2 + 2a_1 + 9a_3) \mod 10$.

The error will **not** be caught if

\[
(7a_1 + 2a_2 + 9a_3) - (7a_2 + 2a_1 + 9a_3) \text{ is a mult. of } 10.
\]

\[
\Rightarrow 5a_1 - 5a_2 \text{ is a mult. of } 10
\]

\[
\Rightarrow 5(a_1 - a_2) \text{ is a mult. of } 10
\]

\[
\Rightarrow a_1 - a_2 \text{ is a mult. of } \frac{10}{5} = 2.
\]

Multiples of 2 are $\pm 2, 0, 2, 4, 6, 8, 10, \ldots$

If $a_1 - a_2 = 0$, then $a_1 = a_2$ so no real error was made.

If $|a_1 - a_2| = 2$, then $a_1 + a_2$ are sep. by 2 units

\[
\Rightarrow 0+2, 1+3, 2+4, 3+5, 4+6, 5+7, 6+8, \text{ and } 7+9.
\]

If $|a_1 - a_2| = 4$, then $a_1 + a_2$ are sep. by 4 units

\[
\Rightarrow 0+4, 1+5, 2+6, 3+7, 4+8, \text{ and } 5+9.
\]

If $|a_1 - a_2| = 6$, then $a_1 + a_2$ are sep. by 6 units

\[
\Rightarrow 0+6, 1+7, 2+8, \text{ and } 3+9.
\]

If $|a_1 - a_2| = 8$, then $a_1 + a_2$ are sep. by 8 units ⇒ $0+8$ and $1+9$.*

If $|a_1 - a_2| = 10$, then $a_1 + a_2$ are sep. by 10 units. Digits are never sep.

by more than 9 units, so this and all higher mult. of 2 are impossible.

*Therefore, errors not caught are when two digits are sep. by 2, 4, 6, or 8 units.
**Example**

Let the check digit \(a_3 = (3a_1 + a_2) \mod 11\)

Will this catch all single-digit errors in the first digit?

Correct code: \(a_1, a_2\) (or \(a_1, a_2, a_3\))

Incorrect code: \(e_1, a_2\) (or \(e_1, a_2, a_3\))

So the correct check digit is \((3a_1 + a_2) \mod 11\) and the incorrect check digit is \((3e_1 + a_2) \mod 11\).

The error will NOT be caught if

\[
(3a_1 + a_2) - (3e_1 + a_2) \text{ is a mult. of } 11
\]

\[
\Rightarrow 3a_1 - 3e_1 \text{ is a mult. of } 11
\]

\[
\Rightarrow 3(a_1 - e_1) \text{ is a mult. of } 11
\]

\[
\Rightarrow a_1 - e_1 \text{ is a mult. of } \frac{11}{3}
\]

Mult. of \(\frac{11}{3}\) are \(\pm \frac{1}{3}, \frac{11}{3}, \frac{22}{3}, \frac{44}{3}, \ldots\)

If \(a_1 - e_1 = 0\), then \(a_1 = e_1\) so we did not really make an error.

If \(|a_1 - e_1| = \frac{11}{3}\), then \(a_1 \neq e_1\) are digits sep. by \(\frac{11}{3}\) units. The difference between digits is always an integer, so this and all other non-integer mult. of \(\frac{11}{3}\) are impossible.

If \(|a_1 - e_1| = 11\), then \(a_1 \neq e_1\) are digits sep. by 11 units.

Digits are never sep. by more than 9 units, so this and all higher mult. of \(\frac{11}{3}\) are impossible.

\[\therefore\text{ all single-digit errors in the first digit will be caught.}\]
**Example**

Let's keep the check digit $a_3 = (3a_1 + a_2) \mod 11$

Will this catch all single-digit errors in the second digit?

**Correct code:** $a_1a_2$ (or $a_1a_2a_3$)

**Incorrect code:** $a_1e_2$ (or $a_1e_2a_3$)

So the correct check digit is $(3a_1 + a_2) \mod 11$ and the incorrect check digit is $(3a_1 + e_2) \mod 11$.

The error will **not** be caught if

$$(3a_1 + a_2) - (3a_1 + e_2)$$

is a mult. of 11

$\Rightarrow a_2 - e_2$ is a mult. of 11

Mult. of 11 are $\{0, 11, 22, \ldots\}$

If $a_2 - e_2 = 0$, then $a_2 = e_2$ so we did not really make an error.

If $|a_2 - e_2| = 11$, then $a_2$ and $e_2$ are digits sep. by 11 units.

Digits are never sep. by more than 9 units, so this and all higher mult. of 11 are impossible.

:. all single-digit errors in the second digit will be caught.
Example

Let's keep the check digit \( a_3 = (3a_1 + a_2) \mod 11 \)

Will this catch all transposition errors?

**Correct code:** \( a_1 a_2 \) (or \( a_1 a_2 a_3 \))

**Incorrect code:** \( a_2 a_1 \) (or \( a_1 a_2 a_3 \))

So the correct check digit is \((3a_1 + a_2) \mod 11\) and

the incorrect check digit is \((3a_2 + a_1) \mod 11\).

This error will **not** be caught if

\[
\frac{(3a_1 + a_2) - (3a_2 + a_1)}{11}
\]

\[
= 2a_1 - 2a_2
\]

is a multiple of 11

\[
= 2(a_1 - a_2)
\]

is a multiple of 11

\[
= a_1 - a_2
\]

is a multiple of \( \frac{11}{2} \).

Multiples of \( \frac{11}{2} \) are \( \pm 0, \frac{11}{2}, 11, \frac{33}{2}, 22, \ldots \).

If \( a_1 - a_2 = 0 \), then \( a_1 = a_2 \) so we did not really make an error.

If \( |a_1 - a_2| = \frac{11}{2} \), then \( a_1 + a_2 \) are digits separated by \( \frac{11}{2} \). Since the difference between digits is always an integer, this and all other non-integer multiples of \( \frac{11}{2} \) are impossible.

If \( |a_1 - a_2| = 11 \), then \( a_1 + a_2 \) are digits separated by 11 units.

Digits are never separated by more than 9 units, so this and all higher multiples of 11 are impossible.

\[
\therefore \text{all transposition errors will be caught}
\]

*Notice that this scheme caught all errors in the first and second digits and all transposition errors. So it caught all single-digit errors & transposition errors.*

It will catch everything unless we make an error on both digits (other than transposition).