Universal Turing Machine
A TM that can simulate the work of all possible Turing Machines.

A universal Java Program is a program that can simulate the work of any other Java Program

\[ \text{process (program } p, \text{ data } d) \]

In compiler!

written in:
Java, JVM
TM

simulates:
Java
TM

called:
Compiler
Universal Turing Machine
our homework

\[ \text{NP} \quad P \subseteq \text{NP} \]

Is \( P = \text{NP} \)? Open problem (unsolved)

What we do know:
There are problems which are harder than any other problem from \( \text{NP} \) they are called \( \text{NP} \)-hard.

Any other problem from \( \text{NP} \) can be reduced to this problem

\[
a z^2 + b z + c = \emptyset
\]

We know: \( a / b, e \)

Find \( z = -b \pm \sqrt{b^2 - 4ac} \)

\[
p \leq \sqrt{\frac{3}{\log t^2}} \Rightarrow 4t^2 + b + t + c = 0
\]

\[
t = z^2 \quad a = p \quad b = g
\]

\[
c = -r
\]

\[
z = \sqrt{t}
\]

\[
(\sqrt{t}, y) \rightarrow y' : \text{c}(x, y)
\]

\[
X \xrightarrow{U(t)} y_y \xrightarrow{U(t)} x_y
\]

\[
x = (a, b, c) \quad y = z \quad c(x, y) \equiv (a z^2 + b z + c = \emptyset)
\]

\[
x = (p, q, r) \quad y = z \quad c(x, y) \equiv (p z^2 + q z + r = \emptyset)
\]
\[ U_1(x) = (p, q, r - c) \]

\[ U_2(z) = \sqrt{z} \]

\[ \text{if } c'(U_1(x), y') \text{ then } x = U_1(x) \rightarrow y' = c'(x', y') \]

\[ \text{if } x' = U_1(x) \text{ and } c'(x', y') \text{ then for } y = U_2(y') \]

\[ \text{we have } c'(x, y) \]

\[ \text{if } c'(U_1(x), y') \text{ then } c'(x, U_2(y')) \]

Given \( x \) find \( y \) for which \( y' = lty' \)

Implication truth table

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>A → B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>T</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>F</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>T</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>T</td>
</tr>
</tbody>
</table>

We need \( U_3(y) \)

\[ \text{if } c'(x, y) \text{ then } c'(U_1(x), U_3(y)) \]

\[ U_3(y) = y^2 \]

\[ \text{if } p^2 + q^2 = r \text{ then } at^2 + bt + c = 0 \]

where \( b = t^2 \)
Definition:

$(U_1, U_2, U_3)$ reduces $(c(x,y), P_e)$ to $(c'(x',y'), P'_e)$ when the following properties are satisfied:

1) if $x' = U_2(x)$ and $c'(x', y')$ and $y = U_3(y)$ then $(c(x, y))$

2) if $c(x, y)$ then for $x' = U_1(x)$ and $y' = U_3(y)$ we have $c'(x', y')$

$$x \xrightarrow{c} y$$

$$U_1 \xrightarrow{U_2} U_3$$

$$x' \xrightarrow{c'} y'$$

$$az + \frac{b}{z} = c$$

$$az^2 + b - cz = 0 \quad a \neq 0$$

$$c(x, y) \quad x = (a, b, c)$$

$$y = z$$

$$c'(x, y) = (az + \frac{b}{z} - c)$$

$$c'(x', y') = x' = (p, q, r)$$

$$y' = t$$
\[ c'(x', y') : \text{pt}^2 + \text{qt} + r = 0 \]

\[ U(a, b, c) = (a, -c, b) \]

\[ p = a \]
\[ q = -c \]
\[ r = b \]

\[ U_2(t) = \{ t \mid t \neq \emptyset \} \]

\[ U_3(z) = z \]

\[ c(x, y') \]

\[ \text{if } ax + b = c \]

Then for \( p = a, q = -c, r = b \) we get

\[ pt^2 + qt + r = 0 \]

for \( t = z \)

\[ c'(x', y') \]
\[ y' = U_3(y) \]

\[ a + \frac{t}{c} = 0 \]
\[ a + \frac{t}{c} = 0 \]

\[ a + \frac{t}{c} = 0 \]

\[ a ln(z) + b = c ln(z) + d = c(x, y) \]

Input \( (a, b, c, d) = x \)

\[ y = z \]

To solve:
\[
\frac{(a-c) \ln(z) = d-b}{p \quad t \quad q} \quad p - t = q \equiv c'(x', y),
\]

\[
x' = (p, q) \quad y' = t \quad t = \ln(z)
\]

\[
U_1(a, b, c, d) = (a-c, d-b)
\]

\[
U_2(t) = \exp(t)
\]

\[
U_3(z) = \ln(z)
\]

General idea

1) Based on an instance \( x \) of the original problem, we find an instance \( x' \) of the aux. problem.

2) Once you find the solution \( y' \) to the aux. problem, you compute the solution \( y \) to the original problem.