Assume the expression e_0 has type T and the branch types are $T_1, \ldots, T_n$. Let $TL(T; T_1, \ldots, T_n)$ be the type matched (i.e. the least type T_k such that $T \leq T_k$). Assume T's class tag is m. For T_i, let m_i be its own class tag and M_i be the largest class tag among its children. Therefore,

$$T \leq T_i \Leftrightarrow m \in (m_i, M_i) \tag{1}$$

Assume $m_1 \geq \cdots \geq m_n$. The case branching algorithm is really simple.

Algorithm 1 Case Branching

```
i ← 1
while i ≤ n do
  if m ∈ (m_i, M_i) then
    return T_i
  end if
  i ← i + 1
end while
return Error
```

To prove its correctness, let $TC(T; T_1, \ldots, T_n)$ be the type chosen by the algorithm and $\mathbb{T}$ be the set of all the types. The proposition can be stated formally as following:

$$\forall i \in \mathbb{N}, \forall T_i \in \mathbb{T}, TL(T; T_1, \ldots, T_n) = TC(T; T_1, \ldots, T_n) \quad (\forall T \in \mathbb{T})$$

Proof. I will induct on n.
Base case ($n = 1$):

$$TL(T; T_1) = \begin{cases} T_1, & T \leq T_1 \\ \text{Error}, & Otherwise \end{cases}$$

$$TC(T; T_1) = \begin{cases} T_1, & m \in (m_1, M_1) \\ \text{Error}, & Otherwise \end{cases}$$

According to (1), $TL(T; T_1) = TC(T; T_1)$
Inductive Hypothesis: Assume $\forall T_i \in \mathbb{T}, TL(T; T_1, \ldots, T_n) = TC(T; T_1, \ldots, T_n)$
Inductive Step:
Notice that:

$$m_1 \geq \cdots \geq m_{n+1} \Rightarrow T_i \not\leq T_1 \quad (2 \leq i \leq n+1) \tag{2}$$

If $m \in (m_1, M_1)$ then $TC(T; T_1, \ldots, T_{n+1}) = T_1$. On the other hand, $m \in (m_1, M_1) \Rightarrow T \leq T_1$. According to (2), the rest of the types are no less than T_1, indicating T_1 is T's closest ancestor, i.e. $TL(T; T_1, \ldots, T_{n+1}) = T_1$. Therefore, $TL(T; T_1, \ldots, T_{n+1}) = TC(T; T_1, \ldots, T_{n+1})$.

If $m \notin (m_1, M_1)$ then $TC(T; T_1, \ldots, T_{n+1}) = TC(T; T_2, \ldots, T_{n+1})$. On the other hand, $m \notin (m_1, M_1) \Rightarrow T \not\leq T_1$. According to the definition of TL, $TL(T; T_1, \ldots, T_{n+1}) = TL(T; T_2, \ldots, T_{n+1})$. According to the hypothesis,

$$\begin{aligned} TL(T; T_1, \ldots, T_{n+1}) &= TL(T; T_2, \ldots, T_{n+1}) \\ &= TC(T; T_2, \ldots, T_{n+1}) = TC(T; T_1, \ldots, T_{n+1}) \end{aligned}$$

□