false, the situation where the requested address were not in memory would need to be handled (by generating a page fault and loading the necessary page):

**Paging Address Translation by Direct Mapping**

This method stores the page table in main memory and the address of this table in the process control block, in a register called the page table base register. Let the page table base register be called \( pt\text{\_base\_register} \), and let memory represent the main store of the computer. Then:

\[
\text{function } \text{NL\_map}(\text{logical\_page}, \text{offset}): \text{physical\_address};
\begin{align*}
\text{begin} \\
\qquad \text{NL\_map} & := \text{memory}[\text{pt\_base\_register} + \text{logical\_page}] \times \text{page\_size} + \text{offset}; \\
\text{end} \quad (* \text{NL\_map} *)
\end{align*}
\]

In pictures, here is what is going on:

---

**Paging Address Translation by Associative Mapping**

In this algorithm, \( \text{assoc\_page\_table} \) represents an associative memory. This function can check a type of memory called "associative memory" (or "cache" or "lookaside memory") which stores both a frame number and a page number. The search is done in parallel, and is much faster than a linear (or binary) search. The function returns the frame number associated with its argument:

\[
\text{function } \text{NL\_map}(\text{logical\_page}, \text{offset}): \text{physical\_address};
\begin{align*}
\text{begin} \\
\qquad \text{NL\_map} & := \text{assoc\_page\_table}(\text{logical\_page}) \times \text{page\_size} + \text{offset}; \\
\text{end} \quad (* \text{NL\_map} *)
\end{align*}
\]

**Paging Address Translation with Combined Associative and Direct Mapping**

This combines the above two methods. The array \( \text{page\_table} \) is a small associative store that can hold only a few

---

Last modified at 10:41 pm on Monday, June 7, 1999
This is the most common method, and is used in modern computers with paging.