253 (Ackermann) Function ack of two natural variables is defined as follows.

$$ack\ 0\ 0\ =\ 2$$
$$ack\ 1\ 0\ =\ 0$$
$$ack\ (m+2)\ 0\ =\ 1$$
$$ack\ 0\ (n+1)\ =\ ack\ 0\ n + 1$$
$$ack\ (m+1)\ (n+1)\ =\ ack\ m\ (ack\ (m+1)\ n)$$

(a) Suppose that functions and function application are not implemented expressions; in that case $n:= ack\ m\ n$ is not a program. Refine $n:= ack\ m\ n$ to obtain a program.

(b) Find a time bound. Hint: you may use function ack in your time bound.

(c) Find a space bound.

After trying the question, scroll down to the solution.

(a) Suppose that functions and function application are not implemented expressions; in that case $n{:=}\ ack\ m\ n$ is not a program. Refine $n{:=}\ ack\ m\ n$ to obtain a program.

§ $n{:=}\ ack\ m\ n \ \Leftarrow$

if $m{=}n{=}0$ **then** $n{:=}\ 2$
else if $m{=}1 \land n{=}0$ **then** $n{:=}\ 0$
else if $n{=}0$ **then** $n{:=}\ 1$
else if $m{=}0$ **then** $n{:=}\ n{-}1.\ \ n{:=}\ ack\ m\ n.\ \ n{:=}\ n{+}1$
else $n{:=}\ n{-}1.\ \ n{:=}\ ack\ m\ n.\ \ m{:=}\ m{-}1.\ \ n{:=}\ ack\ m\ n.\ \ m{:=}\ m{+}1$
fi fi fi fi

Here are the first few values of this function.

	$n=$	0	1	2	3	4	5	6	
$m=$	0	2	3	4	5	6	7	8	$2{+}n$
	1	0	2	4	6	8	10	12	$2{\times}n$
	2	1	2	4	8	16	32	64	2^n
	3	1	2	4	16	65536	*		*tower n*

The entry marked * has about 20000 digits in it, and *tower n* means "two to the power two to the power two to the power ..." with n "two"s. Here is another way to create the table. The top row is 2 3 4 5 and so on; the left column is 2 0 1 1 1 1 and so on; to find an interior item, look left one place, and that's the column number, one row up, to copy from. Just copying; no arithmetic. For example, suppose we want to determine the value of $ack\ 3\ 3$. Look to the left of position $3\ 3$ and you see 4 . So look in the previous row (row 2) under column 4 , and you see 16 . So $ack\ 3\ 3 = 16$.

(b) Find a time bound. Hint: you may use function *ack* in your time bound.

§ For a time bound, we want a function f such that

$t' \le t + f\ m\ n\ \land\ n' = ack\ m\ n\ \land\ m'{=}m\ \Leftarrow$

if $m{=}n{=}0$ **then** $n{:=}\ 2$
else if $m{=}1 \land n{=}0$ **then** $n{:=}\ 0$
else if $n{=}0$ **then** $n{:=}\ 1$
else if $m{=}0$
then $n{:=}\ n{-}1.\ \ t{:=}\ t{+}1.\ \ t' \le t + f\ m\ n\ \land\ n' = ack\ m\ n\ \land\ m'{=}m.$
$n{:=}\ n{+}1$
else $n{:=}\ n{-}1.\ \ t{:=}\ t{+}1.\ \ t' \le t + f\ m\ n\ \land\ n' = ack\ m\ n\ \land\ m'{=}m.$
$m{:=}\ m{-}1.\ \ t' \le t + f\ m\ n\ \land\ n' = ack\ m\ n \land m'{=}m.\ m{:=}\ m{+}1$
fi fi fi fi

In the last alternative, I put $t{:=}\ t{+}1$ before the first recursive call, but not before the second. The one occurrence ensures that every loop includes a time increment. But I could have put another one in. Using Refinement by Cases, and throwing away the unnecessary pieces, we need f to satisfy five things.

$t' \le t + f\ m\ n\ \Leftarrow\ m{=}n{=}0 \land t'{=}t$
$t' \le t + f\ m\ n\ \Leftarrow\ m{=}1 \land n{=}0 \land t'{=}t$
$t' \le t + f\ m\ n\ \Leftarrow\ m{>}1 \land n{=}0 \land t'{=}t$
$t' \le t + f\ m\ n\ \Leftarrow\ m{=}0 \land n{>}0 \land t' \le t + 1 + f\ m\ (n{-}1)$
$t' \le t + f\ m\ n\ \Leftarrow\ m{>}0 \land n{>}0 \land t' \le t + 1 + f\ m\ (n{-}1) + f\ (m{-}1)\ (ack\ m\ (n{-}1))$

Simplifying,

$f\ m\ 0\ \ge\ 0$
$f\ 0\ (n{+}1)\ \ge\ f\ 0\ n + 1$
$f\ (m{+}1)\ (n{+}1)\ \ge\ f\ (m{+}1)\ n + f\ m\ (ack\ (m{+}1)\ n) + 1$

These are the constraints on f . So replace $\ge$ by $=$ and we have a definition of f that gives the exact execution time (in terms of *ack*).

(c) Find a space bound.

no solution given