1. Give a recursive definition of the sequence $\{a_n\}$, $n = 1, 2, 3, \ldots$ if

**a)** $a_n = 6n$.
**b)** $a_n = 2n + 1$.
**c)** $a_n = 10^n$.
**d)** $a_n = 5$.

(a) $\begin{cases} a_1 = 6 \\ a_n = a_{n-1} + 6, \ n > 1 \end{cases}$

(b) $\begin{cases} a_1 = 3 \\ a_n = a_{n-1} + 2, \ n > 1 \end{cases}$

(c) $\begin{cases} a_1 = 10 \\ a_n = 10 \cdot a_{n-1}, \ n > 1 \end{cases}$

(d) $\begin{cases} a_1 = 5 \\ a_n = a_{n-1}, \ n > 1 \end{cases}$

2. Give a recursive definition of

**a)** the set of even integers.
**b)** the set of positive integers congruent to 2 modulo 3.
**c)** the set of positive integers not divisible by 5.

(a) $S = \{0, \pm 2, \pm 4 \ldots\}$

* Basis step: $0 \in S$
* Recusive step:

若 $x \in S$, 则 $x+2 \in S$ 且 $x-2 \in S$ *

(b) $S = \{2, 5, 8, 11 \ldots\}$

$n \equiv 2 \pmod 3$

* Basis step: $2 \in S$
* Recusive step:

若 $x \in S$, 则 $x+3 \in S$ *

(c) $S = \{1, 2, 3, 4, 6, 7, 8, 9, 11 \ldots\}$

* Basis step: $1 \in S$
* Recusive step:

若 $x \in S$, 则 $x+1$

计不会被5整除, $x+1$ 加入 $S$.

else 跳过, $x+2$ 加入 $S$ *

3. Let $S$ be the set of positive integers defined by

*Basis step:* $5 \in S$.

*Recursive step:* If $n \in S$, then $3n \in S$ and $n^2 \in S$.

**a)** Show that if $n \in S$, then $n \equiv 5 \pmod{10}$. (5、15、25...)

**b)** Show that there exists an integer $m \equiv 5 \pmod{10}$ that does not belong to $S$.

(b) check $x$ 是否可由 $3n=x$, $n^2=x$ 造出

當 $x=35$ 時

$3n=35$

$n \doteq 11.67$ (X) not int

$n^2=35$

$n=\sqrt{35} \doteq 5.916...$ (X) not int

∴ $35 \equiv 5 \pmod{10}$, But $35 \notin S$ ※

(a)

※ Statement $P(n)$

※ Basis step:

$5 \in S$, $5 \bmod 10 = 5$

原式成立.

※ Inductive step:

Inductive Step: 假設某 $k \in S$, 且 $k \equiv 5 \bmod 10$

則 $3k \equiv 5 \bmod 10$ → by IH $k \equiv 5 \pmod{10}$ → $3 \cdot k = 3 \cdot 5 = 15 \equiv 5 \pmod{10}$ 成立!

$k^2 \equiv 5 \bmod 10$ → by IH $k \equiv 5 \pmod{10}$ → $k \cdot k = 5 \cdot 5 \equiv 25 \equiv 5 \pmod{10}$ 成立!

※ Conclusion:

由數學歸納法, 可 $n \in S$, then $n \equiv 5 \pmod{10}$ //

4. Give a recursive algorithm for computing $nx$ whenever $n$ is a positive integer and $x$ is an integer, using just addition.

```
Procedure multiply (n: positive integer, x: integer)
    if n=1 then return x
    else return x+multiply(n-1, x)
```
//

5. Devise a recursive algorithm for computing the greatest common divisor of two nonnegative integers $a$ and $b$ with $a < b$ using the fact that $\gcd(a, b) = \gcd(a, b - a)$.

Procedure gcd( a, b: nonnegative integers with a < b)

if a = 0 then return b

else if b = 0 then return a

else if a = b then return a

else if a < b then return gcd(a, b − a)

else return gcd(b, a − b)