RECURSION [Characterized by recurrence or repetition.]

Generally, defining an object in terms of itself is known as recursion.

Recursion can be used to define sequences, functions and sets.

Example :

Consider a sequence of powers of 2, as given by

$$a_n = 2^n \quad \text{for } n = 0, 1, 2 \ldots$$

↓ can be defined as

$a_0 = 2^0 = 1$

$a_1 = 2^1 = 2$

$a_2 = 2^2 = 4$

$a_3 = 2^3 = 8$

$\therefore \; a_{n+1} = 2a_n$

⇓

$a_1 = 2a_0$

$a_2 = 2a_1$

⋮

etc.

A sequence, where the terms of sequence are found from previous terms, is known as recursive sequence.

↓ Once recursive

We can use induction to prove the results about the sequence.

Given $f(0) = 3, \quad f(n+1) = 2f(n) + 3$

so,

$$f(1) = f(0+1) = 2f(0) + 3 = 2 \times 3 + 3 = 9$$

$$f(2) = f(1+1) = 2f(1) + 3 = 2 \times 9 + 3 = 21$$

$$f(3) = f(2+1) = 2f(2) + 3 = 2 \times 21 + 3 = 45$$

$$f(4) = f(3+1) = 2f(3) + 3 = 2 \times 45 + 3 = 93$$

Mathematical Induction

✓ Suppose, we have an infite ladder, and we want to know whether we can reach every step on this ladder

→ ① We can reach the first rung of the ladder

→ ② If we can reach a particular rung of the ladder, then we can reach the next rung.

Example :

Use mathematical induction to show that

proposition

$$P(n) = 1 + 2 + 2^2 + 2^3 + \dots \dots 2^n = 2^{n+1} - 1$$

Basic Step: $P(0) = 2^0 = 1, \quad 2^{0+1} - 1 = 2 - 1 = 1$

Inductive step: We assume that $P(k)$ is true. So,

$$P(k) = 1 + 2 + 2^2 + 2^3 + \dots \dots 2^k = 2^{k+1} - 1$$

Now, we must show that as we assume that $P(k)$ is true, then $P(k+1)$ is true.

Therefore,

$$1 + 2 + 2^2 + \dots \dots + 2^k + 2^{k+1} = (1 + 2 + 2^2 \dots + 2^k) + 2^{k+1}$$

$$= (2^{k+1} - 1) + 2^{k+1}$$

$$= 2 \cdot 2^{k+1} - 1 = 2^{(k+1)+1} - 1$$

$\equiv$ inductive step done

So, we prove that for all non-negative integers the above statement is true

□ Prove sequence using Mathematical Induction Process

Given. $a_1 = 2$, $a_k = 5a_{k-1}$, for $k \geq 2$. It is claimed that terms of the sequence satisfy the equation $a_n = 2.5^{n-1}$, for $n \geq 0$

(1) Write down first three terms —

$a_1 = 2$

$a_2 = 5\,a_{2-1} = 5a_1 = 5 \times 2 = 10$

$a_3 = 5\,a_{3-1} = 5a_2 = 5 \times 10 = 50$

$a_4 = 5\,a_{4-1} = 5a_3 = 5 \times 50 = 250$

(2) To show that every term of the sequence satisfies the equation, we show that the first term of sequence satisfies the equation.

So, the problem statement is

$P(n) \equiv a_n = 2.5^{n-1}$

For $n = 1$. $a_1 = 2$. | $2.5^{n-1} = 2.5^{1-1} = 2.5^0 = 2$

$n =$ $a_2 = 10$ | $2.5^{2-1} = 2.5^1 = 2.5 = 10$

We assume that the statement is true for $n = k$

$a_k = 2.5^{k-1}$

Now, we must show that $a_{k+1} = 2.5^{k+1-1} = 2.5^k$

Recursive

Now, $a_K \equiv$ is the generic term, and is given as–

$$a_K = 5a_{K-1}$$

So, $a_{K+1} = 5a_{K+1-1} = 5a_K = 5\cdot(2\cdot5^{K-1}) = 2\cdot(5\cdot5^{K-1})$

$$= 2\cdot5^K$$

This is the R.H.S to be shown

Fibonacci Numbers:

They are defined as: $f_0 = 0$, $f_1 = 1$, and

$$f_n = f_{n-1} + f_{n-2}$$

So,

$$f_2 = f_{2-1} + f_{2-2} = f_1 + f_0 = 0+1 = 1$$

$$f_3 = f_{3-1} + f_{3-2} = f_2 + f_1 = 1+1 = 2$$

$$f_4 = f_{4-1} + f_{4-2} = f_3 + f_2 = 2+1 = 3$$

RECURSIVELY DEFINED SETS & STRUCTURES

Given $\sum_{k=0}^{a} a_k$, Find the recursive definition —

Basic step: Specify the value of the given fn^c at "0"

So, $$\sum_{k=0}^{0} a_k = a_0$$

Recursive step: Provide a rule for finding the fn^c value at $\overset{n+1}{\underline{\text{an integer}}}$, from its values at $\underbrace{\text{smaller integers}}_{\text{could be "}n\text{"}}$

So, we can write

$$\sum_{k=0}^{n+1} a_k = \left(\sum_{k=0}^{n} a_k\right) + a_{n+1}$$

RECURSIVE SETS

→ When sets are defined recursively

→ Step 1 : Basic Step

→ Step 2 : Recursive step

Basic Step : Initial collection of element is specified.

Recursive Step : Provides rules for forming new elements from those elements that are already in the set.

→ Sometimes, also include an exclusion step/rule

states that elements defined in the basic step and in the recursive step will be in the set.

Given, the set of integer $\mathbb{N}$ define a subset $S \subset \mathbb{N}$

Basic step: Let's assume $3 \in S$.

Recursive step: Now, if $x \in S$, $y \in S$, then $x + y \in S$.

So, $\underbrace{3}_{x} \in S$, $\underbrace{3}_{y} \in S$, then $3 + 3 \in S$

$\equiv 6 \in S$

Again, Now, $3, 6 \in S$. So

$3 + 3,\ 3 + 6,\ 6 + 6$ all belong to "S"

$\equiv 6,\ 9,\ 12$

$\vdots$

So, we can say that S becomes the set of all positive multiples of 3

Using Induction

Along with

Recursive definition

Set "S", as obtained in previous step, is positive multiples of "3"

1. $3 \in S$
2. if $x \in S$ and $y \in S$, then $x + y \in S$
3. No number is in "S" unless it can be shown to be in "S" with the help of 1 & 2

$\hookrightarrow$ Prove it

$\downarrow$

We use mathematical induction.

Proof:

Let's assume that P is the set of positive multiples of 3. ↦ element of the form 3n

We have to show $P \subseteq S$, and $S \subseteq P$

Basic step: When $n=1$, $3.1 \in S$

Induction step: Let's assume that $3k$ is in S.

We want to make sure that $3(k+1)$ is also in S.

$$3(k+1) \equiv 3k+3$$

$3 \in S$

$3k \in S$ ← inductive hypothesis. So,

$3k+3 \in S$

So, $P \subseteq S$

Now, we have to show $S \subseteq P$

We know that, nothing is in "S" unless the rules ① & ② are used at least "n" times.

We use induction on "n"

The integer proved to be in "S" is in fact a positive multiple of 3

Basic step: Apply rule 1.

$3 \in S$, and we see that.

3 is a positive multiple of 3

Inductive step: We assume that ——

Whenever we have $p \in S$, as obtained after "k" or, fewer steps, p is a positive multiple of 3.

Now, at $k+1$ steps –

if $x \in S$, $y \in S$, we have $x+y \in S$

obtained by

$\leq k$ steps $\leq k$ steps

and, both x and y are positive multiples of 3. so,

$x+y$ is also a positive multiple of "3"

That is, $S \subseteq P$

Counting

Product Rule :

If there are n_1 ways to do the first Task

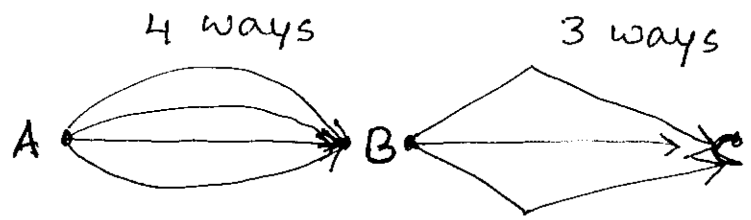
AND

For each of these ways of doing the first task,

there are n_2 ways to do the second task

Then,

There are $n_1 \times n_2$ ways to do the procedure that involves two tasks.



So, there are $4 \times 3 = 12$ ways to reach C from A.

Sum Rule :

If a task can be done either in one of n_1 ways, or in one of n_2 ways,

where none of the set of n_1 ways is the same as any of the set of n_2 ways,

Then, there are $n_1 + n_2$ ways to do the task.

Example :

Assume, ther are three list for a computer project.

List 1 : 23
List 2 : 15
List 3 : 19

No project is on more than one list.

How many possible projects are there to choose form ?

Answer: $23 + 15 + 19$
$= 57$ ways

Inclusion-Exclusion Principle :

Let's assume a task can be done in n_1 or n_2 ways.

But, some of n_1 ways to do the tasks are same as some of the n_2 other ways to do the task.

↓ in this situation

we alsosubtract the number of ways to do the task that ~~are~~ among the n_1 ways and n_2 ways.

Lets rephrase it using sets —

A_1 and A_2 are two sets

$|A_1|$ ways to select an element from A_1

$|A_2|$ ways to select an element from A_2

~~APB~~

So, number of ways to select an element from A_1 or A_2 is :

$$|A_1 \cup A_2| = |A_1| + |A_2| - |A_1 \cap A_2|$$

we subtract the number of ways to select an element common to both sets.

Example:

Consider 350 applicants for a job

250 majored in CS

147 majored in Business

51 majored in CS & Business both

So, How many of these applicants majored neither in CS nor in Business.

Answer:

Using Inclusion - Exclusion Principle, we obtain the number of the students who majored in either CS or Business:

$$|A_1 \cup A_2| = |A_1| + |A_2| - |A_1 \cap A_2|$$

$$= 220 + 147 - \quad = 316.$$

So, $350 - 316 = 34$ of the applicants majored neither in CS nor Business.