Chapter 2
(Part 3)

Recurrence Relation & Recursive Algorithms

Definition

- A **recurrence relation** is an equation that defines a sequence $\{a_n, a_{n+1}, \ldots\}$ based on a rule that gives the next term as a function of one or more of the previous term in the sequence for all integers n with $n \geq n_0$,
  $(a_0,\ a_1, \ldots \quad)$
- A sequence is called a **solution** of a recurrence relation if it terms satisfy the recurrence relation.

Simple Recurrence Relation

The simplest form of a **recurrence relation** is the case where the next term depends only on the immediately previous term.

$$3, 8, 13, 18, 23, \ldots$$

The above sequence shows a pattern, given initial condition, $a_0 = 3$:

$$3,\ 3+5,\ 8+5,\ 13+5,\ 18+5, \ldots$$

$$a_0, a_1, a_2, a_3, a_4, a_5, \ldots$$

generated from an equation:

$$a_n = a_{n-1} + 5, \qquad n \geq 1$$

n_{th} term of a sequence

Recurrence relation can be used to compute any n-*th* term of the sequence, given the initial condition, a_0:

$$a_n = a_{n-1} + 5 \,,\; n \geq 1$$

$$3, 8, 13, 18, 23, 28, a_{n-1} + 5, \ldots$$

$$a_1 = a_0 + 5 \,,\; 3 + 5 = 8$$

$$a_2 = a_1 + 5 \,,\; 8 + 5 = 13$$

$$a_3 = a_2 + 5 \,,\; 13 + 5 = 18$$

$$\vdots$$

$$a_5 = a_4 + 5 \,,\; 23 + 5 = 28$$

Example 1

Consider the following sequence:

3, 9, 27, 81, 243, . . .

The above sequence shows a pattern:

$3^1, \ 3^2, \ 3^3, \ 3^4, \ 3^5, \ldots$ $\qquad$ $3^1, \ 3^2, \ 3^3, \ 3^4, \ 3^5, \ldots$

$a_1, \ a_2, \ a_3, \ a_4, \ a_5, \ldots$ **or** $a_0, a_1, a_2, \ a_3, \ a_4, \ a_5, \ldots$

Recurrence relation is defined by:

$a_n = 3^n, n \geq 1$ **or** $a_n = 3^{n+1}, n \geq 0$

Example 2

Given initial condition, $a_0 = 1$ and recurrence relation:

$$\boldsymbol{a_n = 1 + 2a_{n-1} \;,\; n \geq 1}$$

First few sequence are:

$$a_1 = 1 + 2\,(1) = 3$$
$$a_2 = 1 + 2(3) = 7$$
$$a_3 = 1 + 2(7) = 15$$

1, 3, 7, 15, 31, 63, …

Example 3

Given initial condition, a_0 = 1, a_1 = 2 and recurrence relation:

$$\boldsymbol{a_n = 3(a_{n-1} + a_{n-2}),\ n \geq 2}$$

First few sequence are:

$a_2 = 3(2 + 1) = 9$
$a_3 = 3(9 + 2) = 33$
$a_4 = 3(33 + 9) = 126$

1, 2, 9, 33, 126, 477, 1809, 6858, 26001,…

Example 4

For a photo shoot, the staff at a company have been arranged such that there are 10 people in the front row and each row has 7 more people than in the row in front of it. Find the recurrence relation and compute number of staff in the first 5 rows.

Solution

Notice that the difference between the number of people in successive rows is a constant amount.

This means that the n_{th} term of this sequence can be found using:

$$\boldsymbol{a_n} = \boldsymbol{a_{n-1}} + 7,\ \ n \geq 2 \text{ with } \boldsymbol{a_1} = \mathbf{10}$$

Solution

Number of staff in the first 5 rows:

$\boldsymbol{a_1} = \mathbf{10}$,

$\boldsymbol{a_2 = a_1} + 7$, $10 + 7 = 17$

$\boldsymbol{a_3 = a_2} + 7$, $17 + 7 = 24$

$\boldsymbol{a_4 = a_3} + 7$, $24 + 7 = 31$

$\boldsymbol{a_5 = a_4} + 7$, $31 + 7 = 38$

10, 17, 24, 31, 38

Example 5

Find a recurrence relation and initial condition for

1, 5, 17, 53, 161, 485, …

Solution:

Look at the differences between terms:

4, 12, 36, 108, …

The difference in the sequence is growing by a factor of 3.

4x3, 12x3, 36x3

Solution

However the original sequence is not.

1(3)=**3**, 5(3)=**15**,17(3)=**51**, . . .

1, 5, 17, 53, 161, 485, …

It appears that we always end up with **2 less than** the next term. So, the recurrence relation is defined by:

$a_n = 3(a_{n-1}) + 2$, $n \geq 1$, with initial condition, $a_0 = 1$

$$a_1 = 3(a_0) + 2 = 3(1)+2 = 5$$
$$a_2 = 3(a_1) + 2 = 3(5)+2 = 17$$
$$\vdots$$

Example 6

A depositor deposits RM 10,000 in a savings account at a bank yielding 5% per year with interest compounded annually. How much money will be in the account after 30 years? Let P_n denote the amount in the account after n years.

Solution

Derive the following **recurrence relation**:

$$\boldsymbol{P_n = P_{n-1} + 0.05P_{n-1} = 1.05P_{n-1}}$$

Where, $\boldsymbol{P_n}$ = Current balance and $\boldsymbol{P_{n-1}}$ **=** Previous year balance and 0.05 is the compounding interest.

Solution

Initial condition, P_0 = 10,000. Then,

$P_1 = 1.05P_0$
$P_2 = 1.05P_1 = (1.05)^2 P_0$
$P_3 = 1.05P_2 = (1.05)^3 P_0$
…
$\boldsymbol{P_n = 1.05\mathrm{P}_{n-1} = (1.05)^n P_0}$,

now we can use this formula to calculate n_{th} term without iteration

Solution

Let us use this formula to find P_{30} under the initial condition P_0 = 10,000:

$P_{30} = (1.05)^{30}(10,000) = 43,219.42$

After 30 years, the account contains **RM 43,219.42**.

Example 7

Consider the following sequence:

1, 5, 9, 13, 17

Find the recurrence relation that defines the above sequence.

Answer:

$$a_n = a_{n-1} + 4 \quad , n \geq 1 \text{ with } a_0 = 1$$

Exercise 5

A consumer purchased items costing RM280 with a department store credit card that charges 1.5% interest per month compounded monthly. Write a recurrence relation and initial condition for b_n, the balance of the consumer's account after n months if no further charges occur and the minimum monthly payment of RM25 is made.

Recursive Algorithms

- A recursive procedure is a procedure that invokes itself.
- A recursive algorithm is an algorithm that contains a recursive procedure.
- Recursion is a powerful, elegant and natural way to solve a large class of problems.

example

Factorial problem

- If $n \geq 1$,

$$n! = n\,(n - 1) \ldots 2.\,1$$

  and $0! = 1$

- Notice that, if $n \geq 2$, n factorial can be written,

$$n! = n\,(n - 1)(n - 2) \ldots 2.\,1$$
$$= n.\,(n-1)!$$

example

- n=5
- 5!

$5! = 5. 4. 3. 2. 1 = 5. 4!$

$4! = 4. 3!$

$3! = 3. 2!$

Algorithm

- Input: n, integer ≥ 0
- Output: $n!$
- Factorial (n) {

```
    if (n=0)
        return 1
    return n*factorial(n-1)
}
```

example

- Fibonacci sequence, f_n

$$f_1 = 1$$

$$f_2 = 1$$

$$f_n = f_{n-1} + f_{n-2}, \qquad n \geq 3$$

1, 1, 2, 3, 5, 8, 13, ….

Algorithm

- Input: n
- Output: $f(n)$
- $f(n)$ {

```
    if (n=1 or n=2)
            return 1
    return f (n-1) + f (n-2)
}
```

Example

Consider the following arithmetic sequence: 1, 3, 5, 7, 9,...

Suppose a_n is the term sequence. The generating rule is $a_n = a_{n-1} + 2, n > 1$. The relevant recursive algorithm can be written as

```
f(n)
{   if (n = 1 )
            return 1
    return f(n – 1) +2
}
```

Use the above recursive algorithm to trace n = 4

Solutions

Trace the output if n = 4

f(4)

n = 4
Because n≠1
return f(3) +2

f(4) =7

Return 5 +2

f(3)

n = 3
Because n≠1
return f(2) +2

f(3) =5

Return 3 +2

f(2)

n = 2
Because n≠1
return f(1) +2

f(2) =3

Return 1 +2

f(1)

n = 1
Because n =1
return 1

f(1) =1

Return 1

Answer = 7