Date February 1, 2021

WEEK 8

Exponential Functions

1. ONE-TO-ONE FUNCTIONS

A function $f: A \to B$ is called one-to-one, if for any $x_1 \neq x_2 \in A$, then $f(x_1) \neq f(x_2)$.

Thus, If $f(x_1) = f(x_2)$ $\Rightarrow$ $x_1 = x_2$

\# The Horizontal line test is used to find whether the given fn is one-to-one or not.

\# One-to-one fn never fails horizontal line test.

\# One-to-one fn are ~~not~~ always reversible on their range.

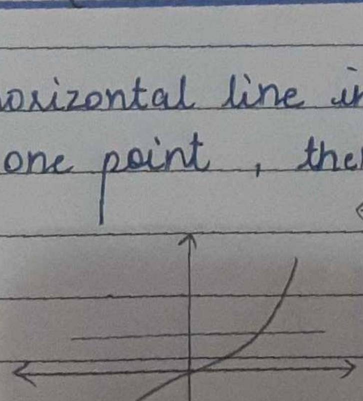
\# If a fn fails the horizontal line test, then it is not reversible.

Examples: $f(x) = x$; $f(x) = x^3$

Theorem (The Horizontal Line Test)

If any horizontal line intersects the graph of a fn f in at most one point, then f is one-to-one.

PROOF: $f(x) = x^3$

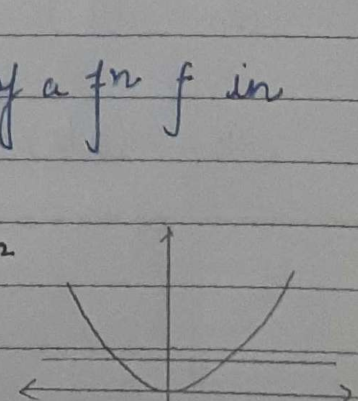


Test Pass.

horizontal line passes through only one pt.

[Let us pass horizontal line, $y = 2$]

$f(x) = x^2$



Test Fail

horizontal line passes through more than 1 pt.

Question : Can we identify the class of functions that are one to one ?

For every $x_1, x_2 \in A$, $x_1 < x_2 \Rightarrow f(x_1) < f(x_2)$
→ Increasing Functions.

For every, $x_1, x_2 \in A$, $x_1 < x_2 \Rightarrow f(x_1) > f(x_2)$
→ Decreasing Functions

Thus, Increasing & Decreasing f^ns are one-to-one.

EXPONENTS

a^r , where $a > 0$ and $r \in \mathbb{R}$

Here, a is base and r is exponent.

Laws of Exponents

For $s, t \in \mathbb{R}$ and $a, b > 0$

Recall, $1^s = 1$; $a^{-s} = \frac{1}{a^s}$; $a^0 = 1,\ a > 0$

(i) $a^s \cdot a^t = a^{s+t}$

(ii) $(a^s)^t = a^{st}$

(iii) $(ab)^s = a^s b^s$

---

2. EXPONENTIAL FUNCTIONS

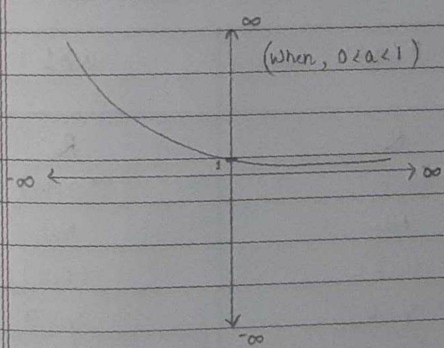
An exponential function in standard form is given by $f(x) = a^x$, where $a > 0$ and $a \neq 1$.

Observations.

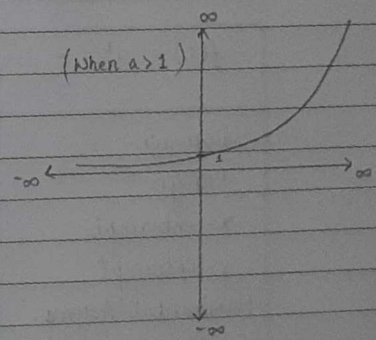
1. Domain of f is $\mathbb{R}$, i.e, $(-\infty, \infty)$
2. Range of f is $\mathbb{R}^+$, i.e, $(0, \infty)$

Why $a \neq 1$?
because, if $a = 1$, ie, $1^x = 1$ (constant f^n)
then it will not remain an exponential function.

GRAPH OF EXPONENTIAL FUNCTIONS



Example, $(1/3)^x$, $(1/5)^x$, $(2/8)^x$, etc



Examples, 2^x, 3^x, 5^x etc.

Properties : $f(x) = 2^x$
→ $\text{Domain}_f = (-\infty, \infty)$
→ $\text{Range}_f = (0, \infty)$
→ y intercept = (0, 1)
→ x intercept = NIL
→ $y = 0$ Horizontal Asymptote

→ End Behaviour :
$x \to \infty$, $2^x \to \infty$
$x \to -\infty$, $2^x \to 0$
→ No roots
→ Increasing f^n.

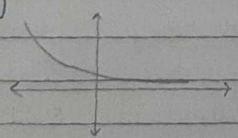
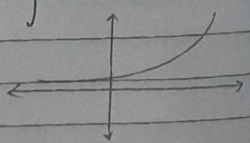
Fact : Every $f(x) = a^x$, $a>1$ has same properties as 2^x.

<u>Properties</u> [$f(x) = 2^{-x} \Rightarrow f(x) = (1/2)^x$] $0<a<1$

- Domain$_f$ = $\mathbb{R}$ → Range$_f$ = $(0, \infty)$
- y intercept = $(0,1)$ → x intercept = NIL (no roots)
- <u>End Behaviour</u>,
  $x \to \infty$, $(1/2)^x \to 0$
  $x \to -\infty$ $(1/2)^x \to \infty$
- It is a decreasing f^n.

Fact : Every $f(x) = a^x$, $0<a<1$ has same properties as $\left(\frac{1}{2}\right)^x$.

SUMMARY

$f(x) = a^x$	$0<a<1$	$a>1$
Domain	R	R
Range	$(0,\infty)$	$(0,\infty)$
x- intercept	NIL	NIL
y- intercept	$(0,1)$	$(0,1)$
Horizontal Asymp.	$y=0$	$y=0$
Inc./Dec. f^n	Decreasing	Increasing
End Behaviour		
$x \to \infty$	$f(x)=a^x \to 0$	$f(x)=a^x \to \infty$
$x \to -\infty$	$f(x)=a^x \to \infty$	$f(x)=a^x \to 0$
Graphs		

3. NATURAL EXPONENTIAL F^N

- From the theory of limits, it is known that
  $$\left(1 + \frac{1}{n}\right)^n \longrightarrow e \quad \text{as } n \longrightarrow \infty$$
- Existence of 'e' is studied in calculus.
- e is irrational no.
- $e \approx 2.71828....$

$\boxed{e \to \text{Euler's no.}}$

<u>Question</u> : Why is 'e' so important ?
→ Interest Rate Calculation
Continuous compounding

<u>Example</u>, Invested = ₹1 Interest = 1%

$$\therefore \left(1 + \frac{0.01}{12}\right)^{12}$$ { If bank revises every month for one year }

Thus, $1 \times e^{0.01t}$ → where t is time

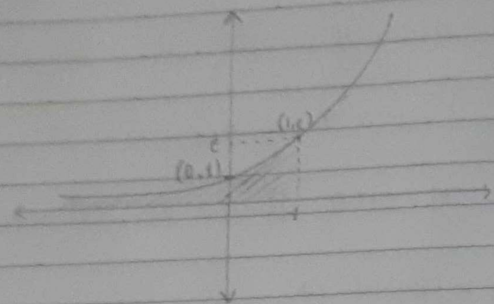
<u>Generalising</u>, $\left(1 + \frac{x}{n}\right)^{nt}$, where t is time, x is interest rate.
$= e^{xt}$

Definition : The natural exponential function is defined as $f(x) = e^x$

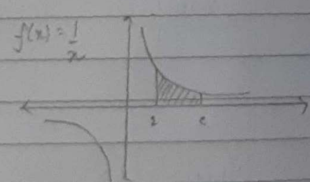
<u>PROPERTIES</u> Domain of f = $\mathbb{R}$
Range of f = $(0, \infty)$

And we know, $e > 1$.

GRAPH :



- e is the slope of the tangent line to $f(x) = e^x$ at $(1, e)$.
- The area under the $f(x) = e^x$ from $(-\infty, 1)$ is e.
- For $f(x) = \frac{1}{x}$, $x \in (1, e)$, the area under the curve is 1 (a unit).



EXAMPLE : Let R be the percent of people who respond to affiliate links under YouTube descriptions & purchase the product in 't' minutes is given by

$$R(t) = 50 - 100\,e^{-0.2t}$$

(a) What is the %age of people responding after 10 min.?
(b) What is the highest percent expected?
(c) How long before $R(t)$ exceeds 30%.!

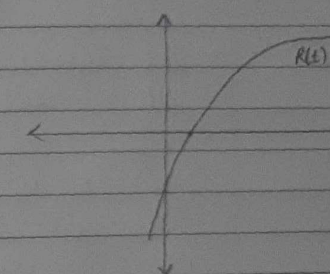
Solution : (a) At $t = 10$,

$$\therefore R(10) = 50 - 100\,e^{-0.2\times 10}$$
$$= 50 - 100\,e^{-2}$$
$$= 50 - \frac{100}{e^2}$$

$$\Rightarrow R(10) = 36.46\,\%$$

(b) $R(t) = 50 - 100\,e^{-0.02t}$

Answer : 50%.



(c) $R(t) = 50 - 100\,e^{-0.02t}$

$\Rightarrow 30 = 50 - 100\,e^{-0.02t}$

$\Rightarrow 100\,e^{-0.02t} = 20$

$\Rightarrow e^{-0.02t} = \frac{1}{5}$ $\quad \ln\left(\frac{1}{5}\right) = -0.02t$

$t \approx 8$ min (from graph)

Domain Of Composite Functions. [f(x), g(x), fog(x)]

Step 1 : Find domain of g
Step 2 : Find range of g
Step 3 : Find domain of f.

CASE 1 → If Range of g $\subseteq$ Domain of f
$\Rightarrow$ Domain of (fog) = Domain of g

CASE 2 → If Range of g $\not\subseteq$ Domain of f, then
We have to eliminate the elements from the domain of g for which we are getting those elements in Range(g) which are not in Domain(f).

Date
February 1, 2021

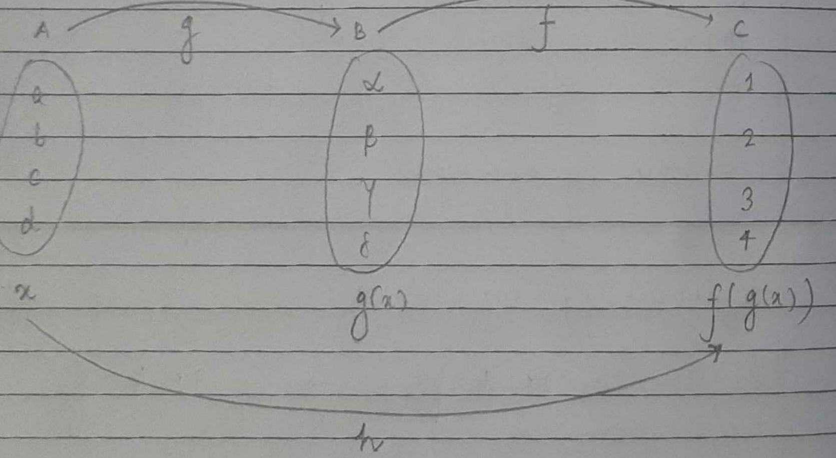
Composite Functions

The composition of functions f & g is denoted by fog & is defined by

$$fog(x) = f(g(x))$$

→ The domain of the composite fn fog is the set of all x such that

- x is in the domain of g
- g(x) is in the domain of f.



∴ $g : A \to B$ Domain : values of x in A
$f : B \to C$ Domain : values of g(x) in B

⇒ $h : A \to C$ Domain : values of x in A, i.e. D_g
$h(x) = f(g(x))$ Range : values of f(g(x)) in C, i.e. R_f.

\# The domain of a composite fn fog(x) is always a subset of the domain of the fn g(x).

\# The range of a composite fn fog(x) is always a subset of the range of the fn f(x).

Examples : Given $f(x) = 3x-4$
$g(x) = x^2$.

Find : (i) fog(x) (ii) gof(x)

(i) $(fog)(x) = f[g(x)] = f[x^2] = 3x^2 - 4$

(ii) $(gof)(x) = g[f(x)] = g[3x-4] = (3x-4)^2$

Q1. Given : $f(x) = x+1$, $g(x) = x^2-1$. Find :
(i) gof(x) (ii) fog(x)

Sol. (i) $gof(x) = g[f(x)] = g[x+1] = (x+1)^2 - 1 = x^2 + 2x$
x^2+1+2x

(ii) $fog(x) = f[g(x)] = f[x^2-1] = x^2-1+1 = x^2$

Q2. Given, $f(x) = 10(x-5)^2 + 100x - 225$ and $g(x) = \sqrt{2x+5}$, find fog(x) and gof(x).

Sol. $fog(x) = f[\sqrt{2x+5}] = 10(\sqrt{2x+5} - 5)^2 + 100(\sqrt{2x+5}) - 225 = 10[(2x+5) + 25 - 10\sqrt{2x+5}] + 100\sqrt{2x+5} - 225$

⇒ $fog(x) = 20x + 50 + 250 - 100\sqrt{2x+5} + 100\sqrt{2x+5} - 225 = 20x + 300 \cancel{- 100\sqrt{2x+5}} - 225$
$= 20x + 75$

$gof(x) = g[10(x-5)^2 + 100x - 225]$

$= \sqrt{2(10x^2 + 250 - \cancel{100x} + \cancel{100x} - 225)} + 5$

$= \sqrt{20x^2 + 50 + 5} = \sqrt{20x^2 + 55}$

Thus, $fog(x) = 20x + 75$ & $gof(x) = \sqrt{20x^2 + 55}$

Q3. Suppose $f(x) = 3x + 10$ & $g(x) = \sqrt{x + 11}$ are two f^n. Find the value of $f(g(5))$. and $g[f(5)]$.

Sol.3. $fog(5) = f[g(5)] = f[\sqrt{5+11}] = f[\sqrt{16}] = f[(\pm 4)]$ Only the value accepted

$f(4) = 3(4) + 10 = 22$ ✓ D_f (square root f^n) $= [0, \infty)$

$f(-4) = -12 + 10 = -2$ ✗

Now, $gof(5) = g[f(5)] = g[15 + 10] = g[25]$

$\Rightarrow g(25) = \sqrt{25 + 11} = \sqrt{36} = 6$

Q4. Give examples, where $fog(x) = gof(x)$.

Sol.4. (i) $f(x) = 7x + 6$ and $g(x) = 4x + 3$

(ii) $f(x) = e^x$ and $g(x) = x$

Determination of Domain for C. f^n.

The following values must be excluded from input x.

If there exists $x \in R$ that is not in the domain of a f^n g, then that x will not be in the domain of some composite f^n (fog).

- if $x \notin D_g \Rightarrow x \notin Dom(fog)$
- $\{x \mid g(x) \notin D_f\}$ must not be included in $Dom(fog)$.

Example: $f(x) = \frac{2}{x-1}$, $g(x) = \frac{3}{x}$

Find $fog(x)$ & Domain (fog).

Sol. $fog(x) = f[g(x)] = f\left[\frac{3}{x}\right] = \frac{2}{3/x - 1} = \frac{2x}{3-x}$

$D_f = R - \{1\}$

$D_g = R - \{0\}$ $\qquad D_{fog} = R - \{3\}$ from here

Thus, Overall : Domain $(fog) = R - \{0, 3\}$

Q5. $f(x) = \frac{1}{x+1}$, $g(x) = \frac{1}{x}$. Find $fog(x)$ & $D(fog)$.

Sol.5. $fog(x) = f\left[\frac{1}{x}\right] = \frac{1}{1/x + 1} = \frac{x}{x+1}$

Rule 1: $D_g = R - \{0\}$

Rule 2: $D_f = R - \{-1\}$ $\therefore g(x) = -1 \Rightarrow 1/x = -1 \Rightarrow x = -1$

$\therefore$ Domain $(fog) = R - \{0, -1\}$

The domain of a composite f^n (fog) is the set of all x such that x is in the domain of f^n g and $g(x)$ is in the domain of a f^n f.

Inverse Functions

The inverse of a function f, f^{-1} is a function such that

$$f^{-1}(f(x)) = x \quad \forall x \in D_f = \text{Range}(f^{-1})$$

$$\& \quad f(f^{-1}(x)) = x \quad \forall x \in D_{f^{-1}} = \text{Range}(f)$$

\# f is one-to-one f^n.

\# All one-to-one f^n are reversible.

Example 1: $g(x) = x^3$ & $g^{-1}(x) = \sqrt[3]{x}$.
Verify that they are inverse of each other.

Sol.

$$gog^{-1}(x) = g[g^{-1}(x)] = g[\sqrt[3]{x}] = g(x^{1/3}) = (x^{1/3})^3 = x$$

$$g^{-1}og(x) = g^{-1}[g(x)] = g^{-1}(x^3) = (x^3)^{1/3} = x$$

Hence verified.

Example 2: Verify f is inverse of g, where $f(x) = \dfrac{x-5}{2x+3}$ and $g(x) = \dfrac{3x+5}{1-2x}$

Sol. Let, $f(x) = y \Rightarrow y = \dfrac{x-5}{2x+3} \Rightarrow (2x+3)y = x-5$
$\Rightarrow 2xy + 3y = x - 5$

$\Rightarrow 3y + 5 = x - 2xy$

$\Rightarrow 3y + 5 = x(1-2y)$

$\Rightarrow x = \dfrac{3y+5}{1-2y} = g(y)$

$\therefore g(x) = f^{-1}(x)$ or $f(x) = g^{-1}(x)$

Other Method: $fog(x) = f[g(x)] = f\left[\dfrac{3x+5}{1-2x}\right]$

$$\Rightarrow fog(x) = \frac{\frac{3x+5}{1-2x} - 5}{2\left(\frac{3x+5}{1-2x}\right) + 3} = \frac{3x+5-5(1-2x)}{6x+10+3(1-2x)}$$

$$\Rightarrow fog(x) = \frac{3x+\cancel{5}-\cancel{5}+10x}{\cancel{6x}+10+3-\cancel{6x}} = \frac{13x}{13} = x$$

Now, $gof(x) = g[f(x)] = g\left[\dfrac{x-5}{2x+3}\right]$

$$\Rightarrow gof(x) = \frac{3\left(\frac{x-5}{2x+3}\right)+5}{1-2\left(\frac{x-5}{2x+3}\right)} = \frac{3x-\cancel{15}+10x+\cancel{15}}{\cancel{2x}+3-\cancel{2x}+10} = \frac{13x}{13} = x$$

$\therefore fog(x) = gof(x) = I(x) = x$

Hence Verified!

Graph of f & f^{-1}

$$x = a \xrightarrow{f} f(a) = y \qquad f(a) = y \xrightarrow{f^{-1}} x = a$$

Values of x changes to y and vice versa.

If $(a, f(a))$ is on the graph of f, then $(f(a), a)$ is on the graph of f^{-1}.

\# Theorem: The graph of f & f^{-1} are symmetric across $y = x$ line

Date February 8, 2021

WEEK 9

Logarithmic Functions

Recall, $f(x) = a^x$ $(a>0, a\neq 1)$ is one-to-one f^n, thus it has an inverse f^n of it.

Definition : The logarithmic function (to the base a) in standard form is

$$y = \log_a(x)$$

and is defined to be the inverse of $f(x) = a^x$.

$$\boxed{y = \log_a x \iff x = a^y}$$

$a^{\log_a x} = x$ and $\log_a a^x = x$

$\text{Domain}(\log_a x) = \text{Range}(a^x) = (0, \infty)$

$\text{Range}(\log_a x) = \text{Domain}(a^x) = \mathbb{R}$

Example : Find the ~~range~~ domain of $f(x) = \log_4(1-x)$

$\text{Domain}(\log_4(1-x)) = \text{Range}(4^{1-x}) = (-\infty, 1)$

Example : Find the domain of $g(x) = \log_3\left(\frac{1+x}{1-x}\right)$

We know, $\text{Domain}(\log_3 x) = (0, \infty)$

i.e, $x \in (0, \infty)$

$\Rightarrow \frac{1+x}{1-x} > 0$

$\Rightarrow 1+x > 0$ & $1-x > 0 \Rightarrow x < 1$

$\Rightarrow x > -1$, Also $x \neq 1$

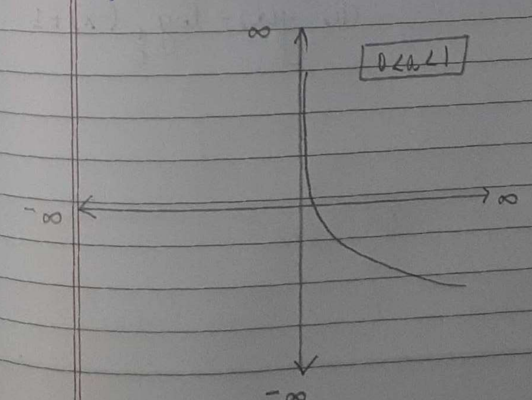
$\therefore$ Domain $= (-1, 1)$ $\cancel{\cup (1, \infty)}$

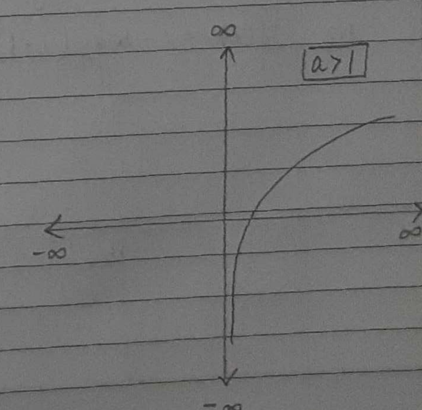
If , $a^u = a^v$ $(a>0, a\neq 1)$ $\Rightarrow u = v$

Question : Find $\log_3\left(\frac{1}{9}\right)$

$\rightarrow \log_3\left(\frac{1}{9}\right) = \log_3\left(3^{-2}\right) = -2$ Ans.

GRAPH OF LOG F^N





Feb 13, 2021

PROPERTIES OF GRAPH OF LOG F^N

Properties of $f(x) = \log_a(x)$

→ $\text{Domain}_f = (0, \infty)$

→ $\text{Range}_f = (-\infty, \infty) = \mathbb{R}$

→ x-intercept = $(1, 0)$

→ y-intercept = NIL

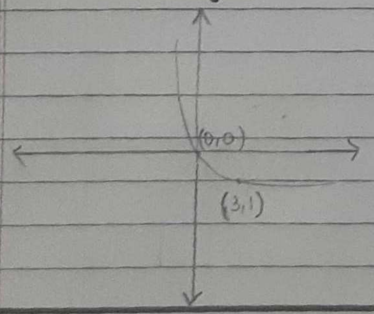
→ Vertical Asymptote at $x = 0$ (y axis)

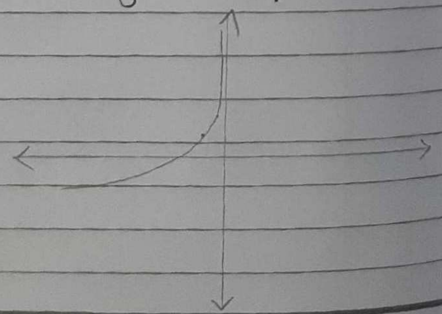
→ f is one to one & passes through $(1,0)$ & $(a, 1)$

→ When, $0 < a < 1$ → f^n is decreasing
and, $a > 1$ → f^n is increasing

EXAMPLE : Draw graphs of following :

(i) $f(x) = -\log_4(x+1)$ (ii) $g(x) = \log_{\frac{1}{4}}(-x) + 1$





NATURAL LOGARITHMIC F^N

The natural logarithmic function is $f(x) = \log_e(x)$ where the base is 'e'.

It is always denoted by $\ln(x)$, i.e,
$f(x) = \ln(x)$

REMARK : $\ln(e^x) = x$, $\forall\ x \in \mathbb{R} = \text{Dom}(e^x)$
$e^{\ln x} = x$, $\forall\ x \in (0, \infty) = \text{Dom}(\ln x)$

COMMON LOG
(BASE 10) $\log x = \log_{10} x$

SOLVING EXPONENTIAL EQ^Ns.

1. Solve for x : $2^{x+1} = 64$

→ $x+1 = \log_2 64 \Rightarrow x+1 = 6 \Rightarrow \boxed{x = 5}$

2. Solve for x : $e^{-x^2} = (e^x)^2 \dfrac{1}{e^3}$

→ $\dfrac{1}{e^{x^2}} = (e^x)^2 \dfrac{1}{e^3}$ $\Rightarrow e^{-x^2} = e^{2x-3}$
$\Rightarrow -x^2 = 2x - 3$
$\Rightarrow x^2 + 2x - 3 = 0$
$\Rightarrow x^2 + 3x - x - 3 = 0$
$\Rightarrow (x+3)(x-1) = 0$; $x = 1, -3$

3. Solve for x : $9^x - 2 \cdot 3^{x+1} - 27 = 0$

Sol. $9^x - 2(3^{x+1}) - 27 = 0$

$\Rightarrow 3^{2x} - 2 \cdot 3^x \cdot 3 - 27 = 0$

$\Rightarrow (3^x)^2 - 6 \cdot (3^x) - 27 = 0$

Let $3^x = y$

$\therefore y^2 - 6y - 27 = 0$

$\Rightarrow y^2 - 9y + 3y - 27 = 0$

$\Rightarrow y(y-9) + 3(y-9) = 0$

$\Rightarrow (y+3)(y-9) = 0$

$\therefore$ When $y = -3$

$\Rightarrow 3^x = -3$

not possible

When $y = 9$

$\Rightarrow 3^x = 9$

$\Rightarrow \boxed{x = 2}$

4. Solve for x : $5^{x-2} = 3^{3x+2}$

Sol. Taking log both sides, $\log 5^{(x-2)} = \log 3^{(3x+2)}$

$\Rightarrow (x-2) \log 5 = (3x+2) \log 3$

$\Rightarrow x \log 5 - 2 \log 5 = 3x \log 3 + 2 \log 3$

$\Rightarrow 3x \log 3 - x \log 5 = -2 \log 3 - 2 \log 5$

$\Rightarrow x [3 \log 3 - \log 5] = -2 [\log 3 + \log 5]$

$\Rightarrow x [\log 27 - \log 5] = -2 \log 15$

$$\Rightarrow x = \frac{-2 \log 15}{\log\left(\frac{27}{5}\right)} = \frac{\log\left(1/225\right)}{\log\left(27/5\right)}$$ Ans.

5. Solve : $x + e^x = 2$

$\rightarrow x + e^x = 2 \Rightarrow e^x = 2 - x \Rightarrow x = \ln(2-x)$

$\Rightarrow \ln(2-x) - x = 0$

$x = 0.443$ [by graph]

PROPERTIES OF LOG FN

1. $\log_a 1 = 0$ $\qquad a \in (0,1)$; $a > 1$

2. $\log_a a = 1$

3. $a^{\log_a x} = \log_a(a^x) = x$

LAWS OF LOGARITHM

Let $r \in \mathbb{R}$, $0 < a < 1$ or $a > 1$; $M, N > 0$

Then,

1. $\log_a(MN) = \log_a M + \log_a N$

2. $\log_a\left(\frac{M}{N}\right) = \log_a M - \log_a N$

3. $\log_a\left(\frac{1}{N}\right) = -\log_a N$

4. $\log_a(M^r) = r \log_a M$

5. $\log_{a^r}(M) = \frac{1}{r} \log_a M$

6. $\log_a x = \frac{1}{\log_x a}$

Application of Laws of Logarithm

1. Simplify : $\log_a\left(\frac{x^3\sqrt{x^2+1}}{(x+3)^4}\right)$

Sol.1 $\log_a x^3 + \log_a \sqrt{x^2+1} - \log_a (x+3)^4$

$= 3\log_a x + \frac{1}{2}\log_a(x^2+1) - 4\log_a(x+3)$

2. Combine using logs : $2\log_a x + \log_a 9 + \log_a(x^2+1) - \log_a 5$

Sol.2. $\log_a x^2 + \log_a 9 + \log_a (x^2+1) - \log_a 5$

$= \log_a\left[\frac{x^2(9)(x^2+1)}{5}\right] = \log_a\left[\frac{9x^4+9x^2}{5}\right]$

CHANGE OF BASE RULE

If $0<a<1$ or $a>1$ and $0<b<1$ or $b>1$

Then, for $x>0$, $\boxed{\log_a x = \frac{\log_b x}{\log_b a}}$

Now, $\log_e x = \frac{\log_{10} x}{\log_{10} e} = \log_{10} x \cdot \log_e 10 = 2.303 \log_{10} x$

Thus, $\boxed{\ln x = 2.303 \log x}$

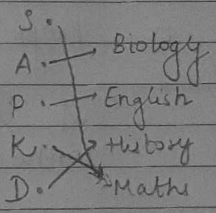
Date February 13, 2021

WEEK 10

Introduction to Graphs

VISUALIZING RELATIONS

- Cartesian Product $A \times B$
  $\{(a,b) \mid a \in A, b \in B\}$
- A relation is a subset of $A \times B$
- Teachers and Courses
  - → T, set of teachers in a college
    C, set of courses being offered
  - → $A \subseteq T \times C$ describes the allocation of teachers to courses
  - → $A = \{(t,c) \mid (t,c) \in T \times C, t \text{ teaches } c\}$



GRAPHS

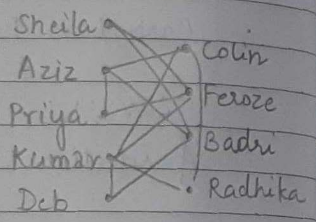
- Graph : $G = (V, E)$
  - → V is a set of vertices or nodes
    - → One vertex, many vertices
  - → E is the set of edges
  - → $E \subseteq V \times V$ – binary relation
- Directed Graph
  - → $(v, v') \in E$ does not imply $(v', v) \in E$

→ The teacher course graph is directed.

- Undirected Graph
  → $(v, v') \in E$ iff $(v', v) \in E$
  → Effectively (v, v'), (v', v) are the same edge
  → Friendship relation

Friendship as a graph

PATHS



- Priya needs some help that Radhika can provide. How will Priya come to know about this?

Priya – Aziz – Badri – Radhika
or Priya – Feroze – Kumar – Radhika

- A path is a sequence of vertices $v_1, v_2, \ldots, v_k$ connected by edges.
  → For $1 \le i < k$, $(v_i, v_{i+1}) \in E$

- Normally, a path does not visit a vertex twice
  → Kumar – Feroze – Colin – Aziz – Priya – Feroze – Sheila
  → Such a sequence is usually called a '<u>walk</u>'.

REACHABILITY

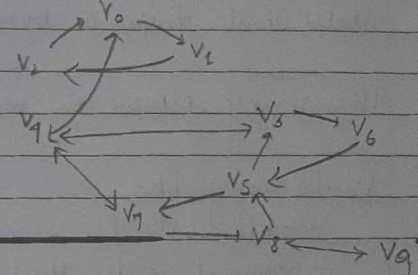
- Paths in directed graphs

- How can I fly from Madurai to Delhi?
  → Find a path from v_9 to v_0.

- Vertex v is reachable from vertex u if there is a path from u to v.

- Typical Questions
  → Is v reachable from u?
  → What is the shortest path from u to v?
  → What are the vertices reachable from u?
  → Is the graph connected? Are all vertices reachable from each other?

AIRLINE ROUTES →



SUMMARY

- A graph represents relationships between entities
  → Entities are vertices / nodes
  → Relationships are edges

- A graph may be directed or undirected
  → A is a parent of B – directed
  → A is a friend of B – undirected

- Paths are sequences of connected edges

- Reachability: is there a path from u to v?

Some General Graph Problems

MAP COLORING

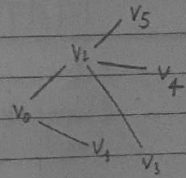
- Assign each state a colour
- States that share a border should be coloured differently
- How many colours do we need?
- Create a graph
  - → Each state is a vertex
  - → connect states that share a border
- Assign colours to nodes so that endpoints of an edge have different colours
- Only need the underlying graph
- Abstraction: if we distort the graph problem is unchanged.

GRAPH COLOURING

- Graph $G = (V, E)$, set of colours C
- Colouring is a f^n $c: V \to C$ such that $(u, v) \in E \Rightarrow c(u) \neq c(v)$
- Given $G = (V, E)$, what is the smallest set of colours need to colour G
  - → 'Four Colour Theorem' for graphs derived from geographical maps, 4 colours suffice
  - → Not all graphs are planar. General case? Why do we care?
- How many classrooms do we need?
  - → Courses and timetable slots
  - → Graph: Edges are overlaps in slots
  - → Colours are classrooms

VERTEX COVER

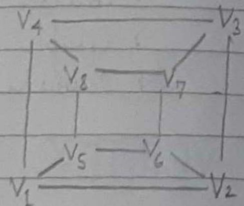
- A hotel wants to install security cameras
  - → All corridors are straight lines
  - → Camera at the intersection of corridors can monitor all these corridor.
- Minimum no. of cameras needed?



- Represent the floor plan as a graph
  - → V - intersections of corridors
  - → E - corridor segments connecting intersections
- Vertex cover
  - → Marking v covers all edges from v
  - → Mark smallest subset of V to cover all edges.

INDEPENDENT SET

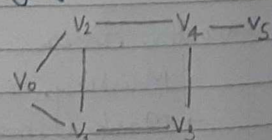
- A dance school puts up group dances
  - → Each dance has a set of dancers
  - → Sets of dancers may overlap across dances
- Organizing a cultural programme
  - → Each dancer performs at most once
  - → Max. no. of dances possible!
- Represent the dances as a graph
  - → V - dances
  - → E - sets of dancers overlap



- Independent set
  - → Subset of vertices such that no two are connected by an edge.

MATCHING

- Class project can be done by one or two people
  - → If two people, they must be friends
- Assume we have a graph describing friendships
- Find a good allocation of groups



- Matching
  - → $G = (V, E)$, an undirected graph
  - → A matching is a subset $M \subseteq E$ of mutually disjoint edges
- Find a maximal matching in G
- Is there a perfect matching, covering all vertices?

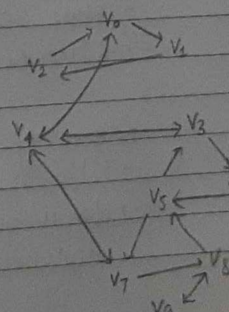
SUMMARY

- Graphs are useful abstract representations for a wide range of problems
- Reachability and connectedness are not the only interesting problems we can solve on graphs
  - → Graph colouring
  - → Vertex cover
  - → Independent set
  - → Matching
  - ....

Working with Graphs

AIRLINE ROUTES

- Looking at the picture of G, we can "see" that V_0 is reachable from V_9



- How do we represent this picture so that we can compute reachability?

ADJACENCY MATRIX

- Let $|V| = n$
  → assume $V = \{0, 1, \dots, n-1\}$
  → use a table to map actual vertex "names" to this set

- Edges are now pairs (i, j), where $0 \le i, j < n$
  → usually assume ~~i≠~~ $i \neq j$, no self loops

- Adjacency matrix
  → Rows and columns numbered $\{0, 1, \dots, n-1\}$
  → $A[i,j] = 1$, if $(i,j) \in E$

- Undirected graph
  → $A[i,j] = 1$ iff $A[j,i] = 1$
  → symmetric across main diagonal

COMPUTING WITH ADJACENCY MAT.

- Neighbours of i - column j with entry 1
  → scan row i to identify neighbours of i
  → Neighbours of 6 are 3 and 5

- Directed graph
  → rows represent outgoing edges
  → columns represent incoming edges

- Degree of a vertex i
  → No. of edges incident on i, degree(6) = 2
  → For directed graphs, outdegree & indegree
  indegree(6) = 1, outdegree(6) = 1

CHECKING REACHABILITY

- Is Delhi (0) reachable from Madurai (9)?

- Mark 9 as reachable

- Mark each neighbour of 9 as reachable

- Systematically mark neighbours of marked vertices

- Stop when 0 becomes marked

- If marking process stops without target becoming marked the target is unreachable

- Mark source vertex as ~~re~~ reachable

- Systematically mark neighbours of marked vertices

- Stop when target becomes marked

- Need a strategy to systematically explore marked neighbours

- Two primary strategies
  → Breadth first : propagate marks in 'layers'
  → Depth first : explore a path till it dies out, then backtrack

ADJACENCY LISTS

- Adjacency matrix has many 0's
  - → size is n^2, regardless of no. of edges
  - → undirected graph : $|E| \le \frac{n(n-1)}{2}$
  - → directed graph : $|E| \le n(n-1)$
  - → Typically $|E|$ is much less than n^2
- Adjacency list
  - → List of neighbours for each vertex

0	{1, 4}	5	{3, 7}
1	{2}	6	{5}
2	{0}	7	{4, 8}
3	{4, 6}	8	{5, 9}
4	{0, 3, 7}	9	{8}

COMPARING REPRESENTATIONS

- Adjacency list typically requires less space
- Is j a neighbour of i ?
  - → Check if $A[i,j] = 1$ in adjacency matrix
  - → Scan all neighbours of i in adjacency list
- Which are neighbours of i ?
  - → Scan all n entries in row i in adjacency matrix
  - → Takes time proportional to (out) degree of i in adj. list
- Choose representation depending on requirement

SUMMARY

- To operate on graphs, we need to represent them
- Adjacency matrix
  - → $n \times n$ matrix, $A[i,j] = 1$ iff $(i,j) \in E$
- Adjacency list
  - → For each vertex i, list of neighbours of i
- Can systematically explore a graph using these representations
  - → For reachability, propagate marking to all reachable vertices.

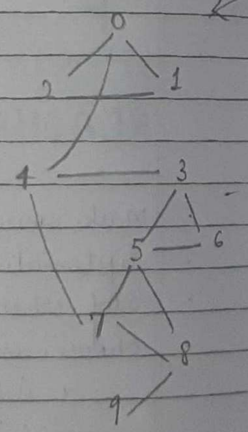
Breadth First Search

REACHABILITY IN A GRAPH

- Mark source vertex as reachable
- Systematically mark neighbours of marked vertices
- Stop when target becomes marked
- Choose an appropriate representation
  - → Adjacency matrix
  - → Adjacency list
- Strategies for systematic exploration
  - → BFS : propagate marks in layers
  - → DFS : explore a path till it dies out, then backtrack

BREADTH FIRST SEARCH (BFS)

- Explore the graph level by level
  - → first visit vertices one step away.
  - → then two steps away
  - → ....
- Each visited vertex has to be explored
  - → Extend the search to its neighbours
  - → Do this only once for each vertex!
- Maintain information about vertices
  - → which vertices have been visited already
  - → among these, which are yet to be explored
- Assume $V = \{0, 1, \dots, n-1\}$



- visited : $V \to \{True, False\}$ tells us whether $v \in V$ has been visited
  - → initially, visited(v) = False for all $v \in V$
- Maintain a sequence of visited vertices yet to be explored
  - → a queue - first in, first out
  - → initially empty
- Exploring a vertex i
  - → For each edge (i, j), if visited (j) is False,
    - · set visited(j) to True
    - · append j to the queue

- Initially
  - → visited (v) = False for all $v \in V$
  - → queue of vertices to be explored is empty
- Start BFS from vertex j
  - → set visited(j) = True
  - → Add j to the queue
- Remove and explore vertex i at head of queue
  - → For each edge (i, j), if visited (j) is False
    - · set visited (j) to True
    - · Append j to the queue
- Stop when queue is empty

EXAMPLE : BFS from Vertex 7

	Visited
0	True
1	True
2	True
3	True
4	True
5	True
6	True
7	True
8	True
9	True

- Mark 7 & add to queue
- Explore 7, visit {4,5,8}
- Explore 4, visit {0,3}
- Explore 5, visit {6}
- Explore 8, visit {9}
- Explore 0, visit {1,2}
- Explore 3
- Explore 6
- Explore 9
- Explore 1
- Explore 2

To Explore Queue

7				
4	5	8		
5	8	0	3	
8	0	3	6	
0	3	6	9	
3	6	9	1	2
6	9	1	2	
9	1	2		
1	2			
2				

ENHANCING BFS TO RECORD PATHS

- If BFS from i sets visited(j) = True, we know that j is reachable from i
- How do we recover a path from i to j?
- visited(j) was set to True when exploring some vertex k
- Record parent(j) = k
- From j, follow parent links to trace back a path to i

<u>Example</u>: BFS from Vertex 7 with parent information

	Visited	Parent
0	True	4
1	True	0
2	True	0
3	True	4
4	True	7
5	True	7
6	True	5
7	True	
8	True	7
9	True	8

- Mark 7, add to queue
- Explore 7, visit {4,5,8}
- Explore 4, visit {0,3}
- Explore 5, visit {6}
- Explore 8, visit {9}
- Explore 0, visit {1,2}
- Explore 3
- Explore 6
- Explore 9
- Explore 1
- Explore 2

To explore Queue				
7				
4	5	8		
5	8	0	3	
8	0	3	6	
0	3	6	9	
3	6	9	1	2
6	9	1	2	
9	1	2		
1	2			
2				

Path from 7 to 6 is 7 - 5 - 6
Path from 7 to 2 is 7 - 4 - 0 - 2

ENHANCING BFS to RECORD DISTANCE

- BFS explores neighbours level by level
- By recording the level at which a vertex is visited, we get its distance from the source vertex
- Instead of visited(j), maintain level(j)
- Initialize level(j) = -1 for all j
- Set level(i) = 0 for source vertex
- If we visit j from k, set level(j) to level(k) + 1
- level(j) is the length of the shortest path from the source vertex, in no. of edges

<u>EXAMPLE</u>: BFS from vertex 7 with parent and distance information

	Level	Parent
0	2	4
1	3	0
2	3	0
3	2	4
4	1	7
5	1	7
6	2	5
7	0	
8	1	7
9	2	8

- Mark 7, add to queue
- Explore 7, visit {4,5,8}
- Explore 4, visit {0,3}
- Explore 5, visit {6}
- Explore 8, visit {9}
- Explore 0, visit {1,2}
- Explore 3
- Explore 6
- Explore 9
- Explore 1
- Explore 2

To Explore Queue				
7				
4	5	8		
5	8	0	3	
8	0	3	6	
0	3	6	9	
3	6	9	1	2
6	9	1	2	
9	1	2		
1	2			
2				

SUMMARY

- Breadth first search is a systematic strategy to explore a graph, level by level.
- Record which vertices have been visited
- Maintain visited but unexplored vertices in a queue
- Add parent info. to recover the path to each reachable vertex.
- Maintain level of information to record length of the shortest path, in terms of no. of edges.
- In general, edges are labelled with a cost (dist, time, ticket price, ...)
- Will look at weighted graphs, where shortest paths are in terms of cost, not no. of edges

Date February 14, 202

Depth First Search

- Start from i, visit an unexplored neighbour j
- suspend the exploration of i and explore j instead
- continue till you reach a vertex with no unexplored neighbours
- Backtrack to nearest suspended vertex that still has an unexplored neighbour
- Suspend vertices are stored in a stack
  - → Last in, first out
  - → Most recently suspended is checked first

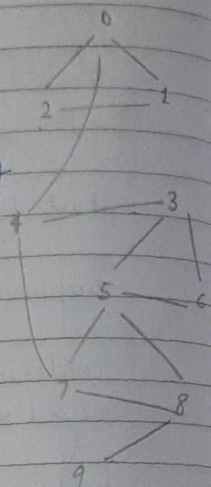
EXAMPLE : DFS from vertex 4

	Visited
0	True
1	True
2	True
3	True
4	True
5	True
6	True
7	True
8	True
9	True

Stack of suspended vertices						
4						
4	0					
4	0	1				
4	0					
4						
4	3					
4	3	5				
4	3	5				
4	3	5	7			
4	3	5	7	8		
4	3	5	7			
4	3	5				
4	3					

- Mark 4, suspend 4, explore 0
- suspend 0, explore 1
- suspend 1, explore 2
- Backtrack to 1, 0, 4
- Suspend 4, explore 3
- Suspend 3, explore 5
- Suspend 5, explore 6
- Backtrack to 5,
- Suspend 5, explore 7
- Suspend 7, explore 8
- Suspend 8, explore 9
- Back track to 8, 7, 5, 3, 4

- Paths discovered by BFS are not shortest paths, unlike BFS



- Useful features can be found by recording the order in which DFS visits vertices
- DFS numbering - maintain a counter
  - → increment and record counter value each time you start and finish exploring a vertex
- DFS numbering can be used to
  - → Find cut vertices (deleting vertex disconnects graph)
  - → Find bridges (deleting edge disconnects graph)

SUMMARY

- DFS is another systematic strategy to explore a graph
- DFS uses a stack to suspend exploration and move to unexplored neighbours
- DFS numbering can be used to discover many facts about graphs

---

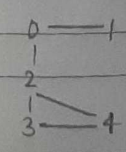
Applications of BFS & DFS

- BFS and DFS systematically compute reachability in graphs
- BFS works level by level
  - → Discovers shortest paths in terms of no. of edges
- DFS explores a vertex as soon as it is visited neighbours
  - → Suspend a vertex while exploring its neighbours
  - → DFS numbering describes the order in which vertices are explored
- Beyond reachability, what can we find out about a graph using BFS/DFS?

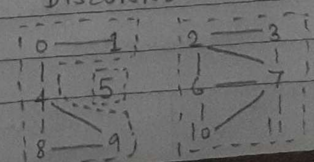
CONNECTIVITY

- A undirected graph is connected if every vertex is reachable from every other vertex.
- In a disconnected graph, we can identify the connected components
  - → Maximal subsets of vertices that are connected
  - → Isolated vertices are trivial components

CONNECTED GRAPH



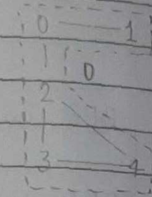
DISCONNECTED GRAPH



IDENTIFYING CONNECTED COMPONENTS

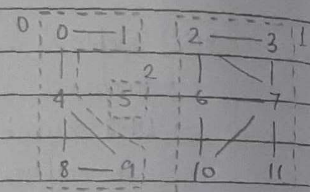
- Assign each vertex a component no.

Connected Graph



- Start BFS/DFS from vertex 0
  - → Initialise component no. to zero
  - → All visited nodes form a connected component
  - → Assign each visited node component number 0.

Disconnected Graph

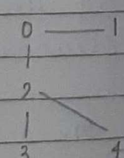


- Pick smallest unvisited node j
  - → increment component no. to 1
  - → Run BFS/DFS from node j
  - → Assign each visited node component no. 1

- Repeat until all nodes are visited.

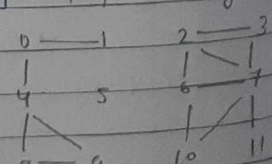
DETECTING CYCLES

Acyclic Graph



- A cycle is a path (technically, a walk) that starts & ends at the same vertex
  - → 4-8-9-4 is a cycle
- → Cycle may repeat a vertex :
  - 2-3-7-10-6-7-2
  - → cycles should not repeat edges : i-j-i is not a cycle, i.e, 2-4-2
  - → simple cycle – only repeated vertices are start & end.

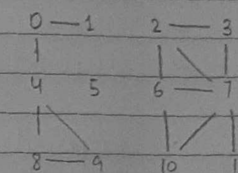
Graph with cycles



- A graph is acyclic if it has no cycles.

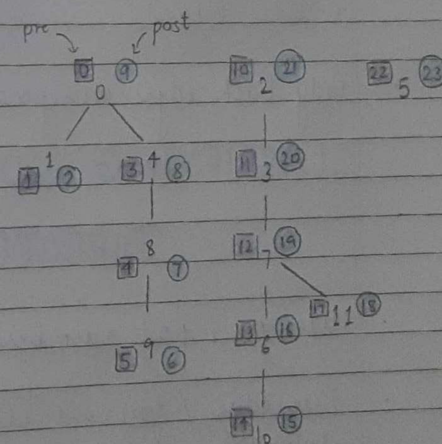
BFS tree

- A tree is a minimally connected graph
- Edges explored by BFS form a tree
  - → Technically, one tree per component
  - → Collection of trees is a forest

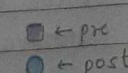


- Facts about trees
  - → A tree on n vertices has n-1 edges
  - → A tree is acyclic
- Any non tree edge creates a cycle
  - → detect cycles by searching for non tree edges

DFS tree



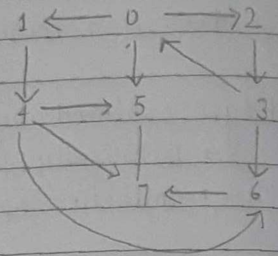
- Maintain a DFS counter, initially 0
- Increment counter each time we start a finish exploring a node
- Each vertex is assigned an entry number (pre) and exit no. (post)



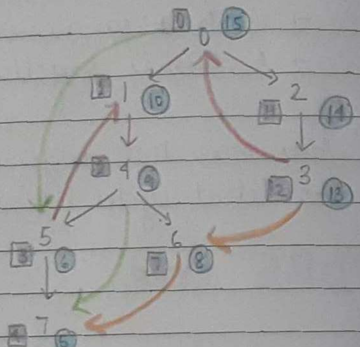
- As before non-tree edges generates cycles

DIRECTED CYCLES

- In a directed graph, a cycle must follow same direction
  - → 0 → 2 → ~~0~~3 → 0 is a cycle
  - → 0 → 5 → 1 ← 0 is not



- Tree Edges
- Different type of non tree edges
  - → Forward Edges
  - → Back Edges
  - → Cross Edges



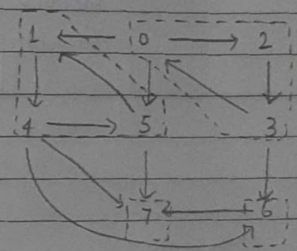
- Only back edges correspond to cycles

CLASSIFYING NON-TREE EDGES IN DIRECTED GRAPHS

- Use pre / post numbers
- Tree edge / forward edge (u,v)
  Interval [pre(u), post(u)] contains [pre(v), post(v)]
- Back Edge (u,v)
  Interval [pre(v), post(v)] contains [pre(u), post(u)]
- Cross Edge (u,v)
  Interval [pre(u), post(u)] and [pre(v), post(v)] are disjoint.

CONNECTIVITY IN DIRECTED GRAPHS

- Take directions into account
- Vertices i and j are strongly connected if there is a path from i to j and path from j to i
- Directed graphs can be decomposed into strongly connected components (SCC's)
  - → within an SCC, each pair of vertices is strongly connected



- DFS numbering can be used to compute SCC's.

SUMMARY

- BFS and DFS can be used to identify connected components in an undirected graph
  - → BFS and DFS identify an underlying tree, non-tree edges generate cycles.
- In a directed graph, non-tree edges can be forward / back / cross
  - → only back edges generate cycles
  - → classify non-tree edges using DFS numbering

- Directed graphs decompose into strongly connected components
  - DFS numbering can be used to compute SCC decomposition
- DFS numbering can also be used to identify other features such as articulation points (cut vertices) and bridges (cut edges)
- Directed acyclic graphs are useful for representing dependencies
  - Given course prerequisites, find a valid sequence to complete a programme.