

23/08/23

Quizes → 4 PF
Lec #02

Program analysis chart

Book → How to Programs (7th Edition)

By Paul Deitel

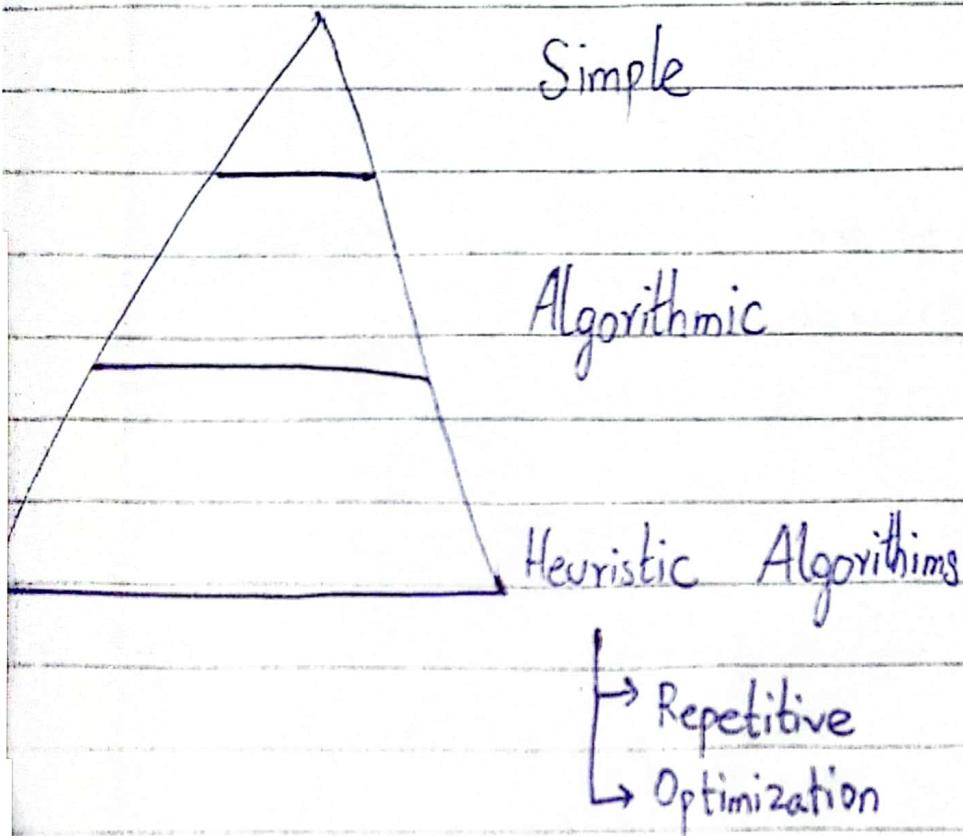
Program & Problem solving and
program design

Problem solving:

- ① Identifying the Problem:
- ② Understanding the Problem:
- ③ Alternatives solution
- ④ ~~Best~~ Best solution
- ⑤ Listing instructions
- ⑥ Testing : Result and Evaluation

25/08/23 PF

Lec#03



PAC
RSOF, IPO

↓
→ Next

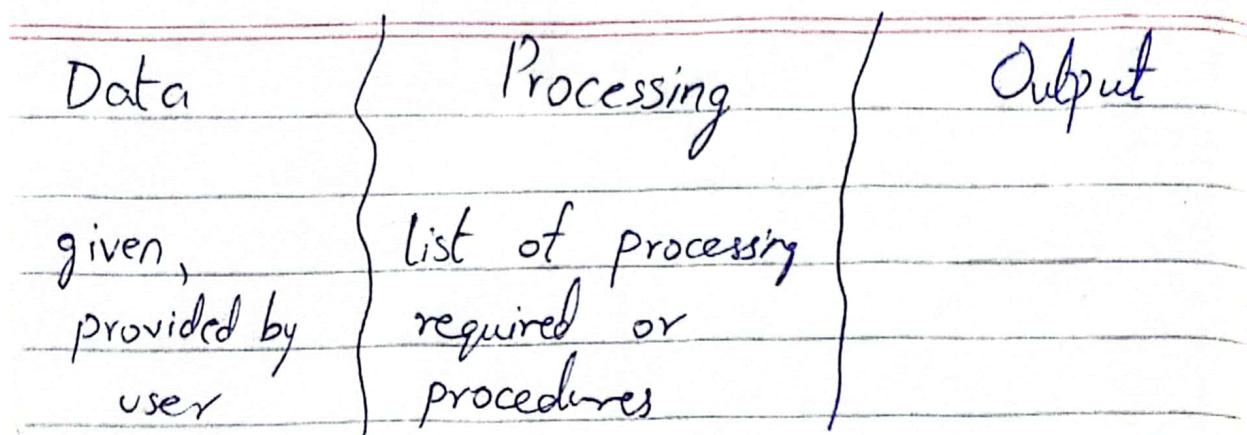
Pre-Programming Phase

28/08/23

Lec#04

- * PAC → Program Analysis Chart
- * IPO
- * Program Flowcharts
- * Algorithms

PAC



Q. Draw PAC for
 An HD 720p photo takes 5mb
 Space on a memory card. How
 many photos can be stored on
 a 1 GB card?

Data	Processing	Output
1 HD 720p photo takes 5mb. $1 \text{ GB} = 1024 \text{ mb}$	* $1024 \div 5 = \text{No. of photos}$ * Round off	No. of photos

$$\text{No. of photos} =$$

card_storage/photo_storage

Data	Processing	Output
resolution = 720p one_photo = 5mb storage = 1 GB	* Total_storage = storage * 1024 * $\text{No. of photos} = \text{total_storage}/\text{one_photo}$	* $\{\text{No. of photos}\}$ can be stored on a 1GB card.

PF : Lec#04 : 28/03/23

Program Analysis Chart (PAC)

Q. Ali wants to calculate amount of cat-food he should purchase for each month. He has 3 cats and each cat eats 500gm in 5 days.

PAC

Data:

months = 12

cats = 3

per-cat = 500gm

days = 5

days permonth = 30

Processing:

per-day = per-cat / days

cat-permonth = days_permonth * per-day

cat-food = cat-permonth * cats

Output:

Ali ha

to buy

{cat-food

for each

month

IPO

Input:

Processing:

Module:

- cats

- per-day = per-cat / days

1000

- per-cat

- cat-permonth = days_permonth

1001

- days_permonth = 30

* per-day

- days = 5

- cat-food = cat_permonth * cats

1002

- Print cat-food

1003

- End

0000

Output:
cat food

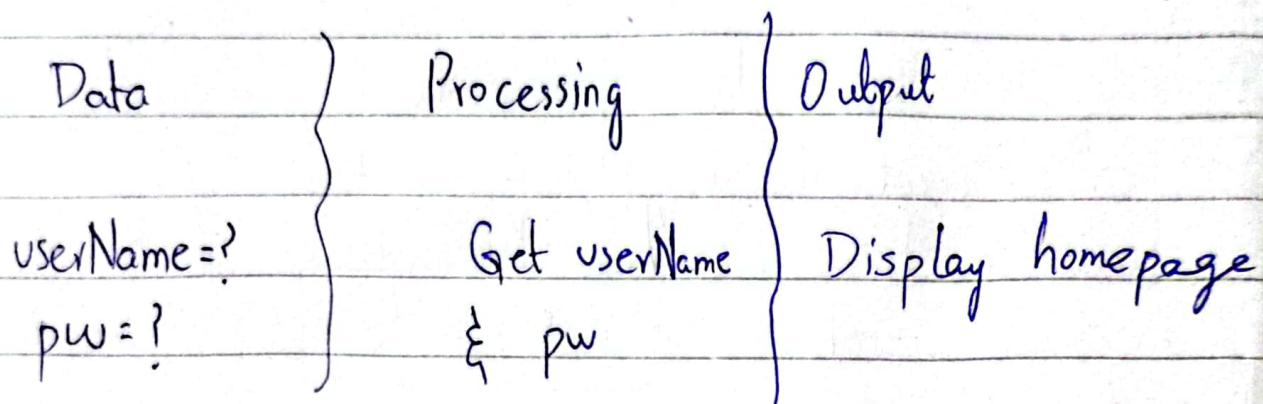
*Validation

PF

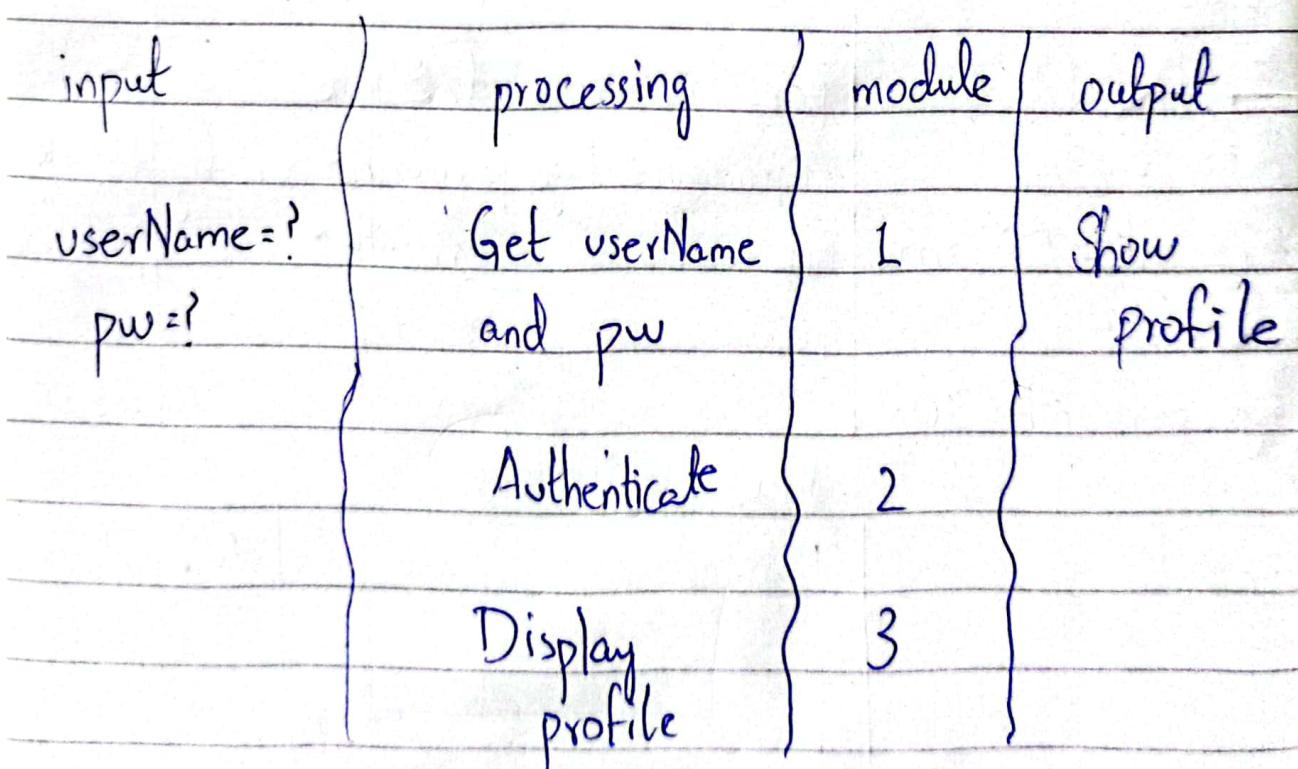
lect#05 : 29/08/23

Login to Facebook:

PAC



IPO

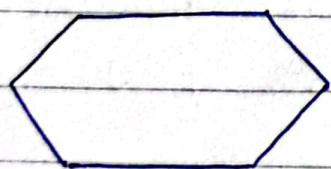


*Universal language of programming:
(Flowchart & Algorithm)

Flowchart:

- Graphical representation of steps or actions to implement a particular module
- Improve processes by identifying bottlenecks and defects.
- Communicate/Explain a process
- Visualize complex processes
- Independent of any programming language.

Symbols:



: Used for doing a repetition or looping of certain steps

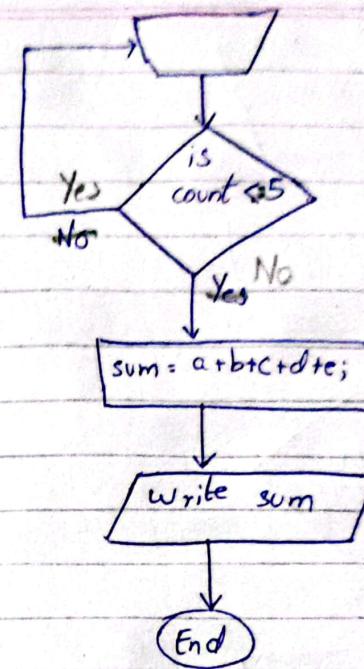
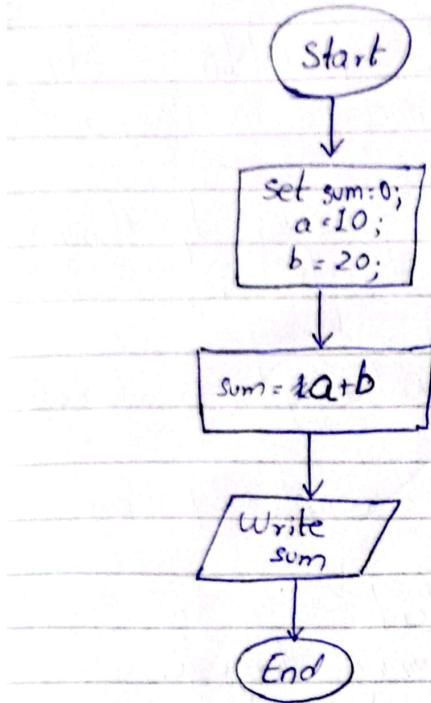


: connect on same page

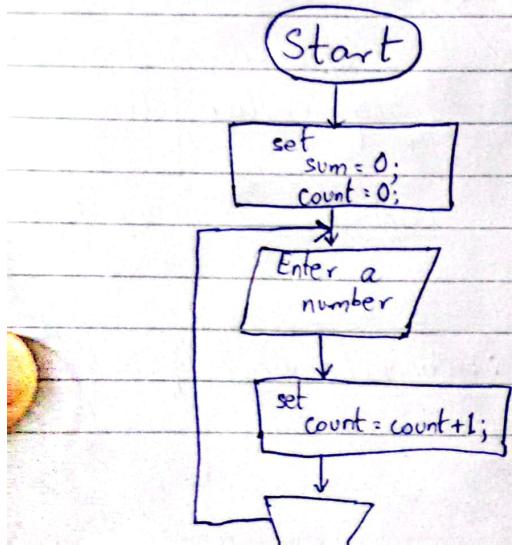


: connect on different/next page

Q. Add 10 and 20



Q. Find sum of 5 numbers



Friday
of lecture 6 1st / 09 / 23

Algorithm:

Well defined sequential computational technique, accepts values as input and produces output needed to solve a problem.

→ sequential structure of program

→ Descriptive form of an IPO chart

→ Formulated way of flow.

pseudocode

Q. Write an algorithm

to determine a student's final grade and indicate whether it is passing or failing
Required (Input marks of 4 subjects)
Output (Grade:)

1. Start

2. Read marks of four subjects. maths, physics, english

3. Sum marks of all subjects to M+P+E+L obtained marks | ISL obtained marks = $M+P+E+L$

4. Set max marks to 400.

5. Set percentage to obtained marks * 100
max marks

6. If percentage > 50 go to set step 7
else step 8.

7. Write "Grade: Pass"

8. Write "Grade: Fail"

9. End

logic structures:

Sequential Structure

→ One after another

Decision Structure

One out of → Two possible sets of instructions

loop structure

→ execute many times

Case structure

→ one set of instructions
out of several sets.

- + Improve readability
- + Proper naming of variables
- + Internal documents
- + Proper indentation

$$x = 120 + 30$$

equation:

can have multiple outputs possibilities

expression:

$$120 > 30$$

possibility

can only ^{be} have one out of two outputs: (e.g. True or False)

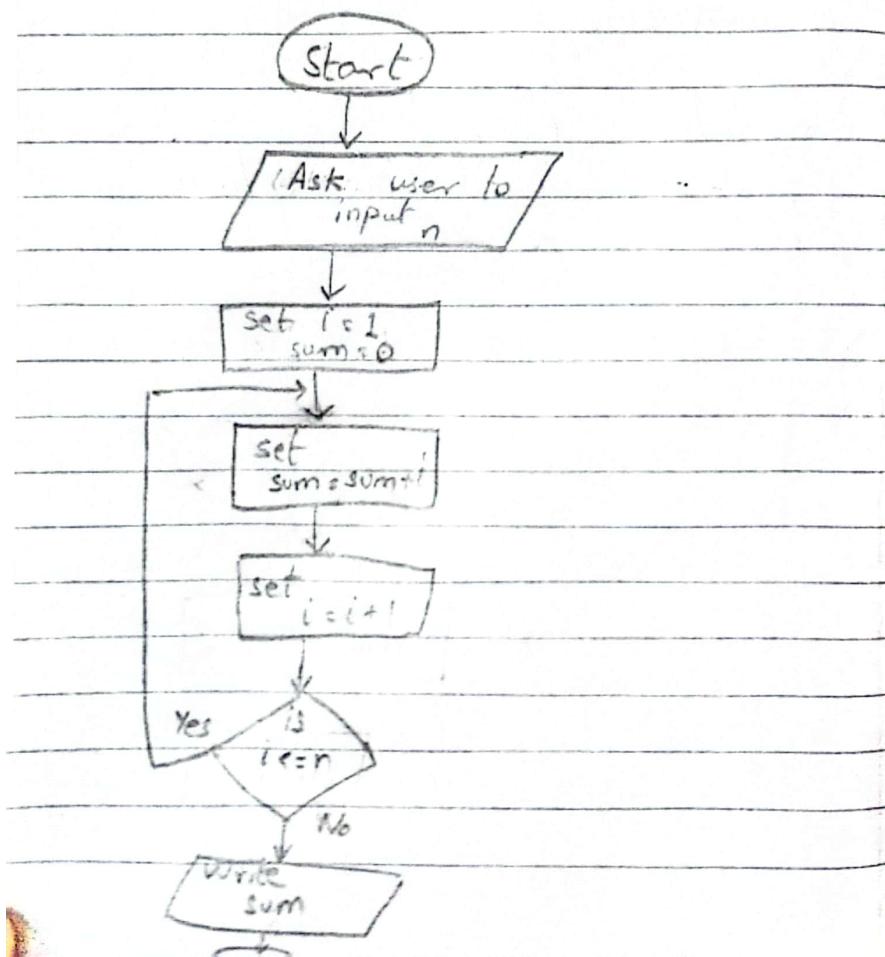
→ Always used in Decision Structures

loop symbol

```
while( ) {  
    }  
    while( );
```



Q. Ask user for a number n and prints the sum of numbers 1 to n .



PF lec#08

5/09/23

Characteristics of C:

- > Small size
- > Extensive use of function calls
- > Loose typing - unlike PASCAL.
- > Structured language
- > low level - (Bitwise) programming readily available
- > Pointer implementation → extensive use of pointers for memory, arrays, structures and functions.

Writing C program:

- > text editor
- > source code
- > source file
- > invoke the c compiler before program can be executed (run).

Stage 1: Preprocessing

Stage 2: Compilation:

Stage 3: linking:

Editor

.c file = program file in high level language.

Compiler

.obj file = object file created by C compiler

Linker:

.exe file = create from object file for execution.

PF

Lec # 11:

12/09/23

Tuesday

Algorithm

VS

Pseudocode

- For any problems
- Exclusively for computer problems
- Step wise description
- Without steps
- Also for non-programmer understanding
- Exclusively for programmer understanding

Garbage value: value assigned ~~to~~ previously
of which reference may
removed.

PF Lect #012

15/09/23 (Friday)

Q. Write a program to identify
if max/min power limit has
been breached by input.

&&
(short circuit[↓] operator)

if one condition is false already, it
will not check the other.

break:

moves execution out of the body

$$\sum_{x=1}^{20} f(x) \rightarrow \text{Submission}$$

↓
starting

$$\prod_{\text{Loops}} \rightarrow \text{Product of}$$

- Counter-control loop: \rightarrow for loop
 others we know
 how many times repetition required, starting point, ending point

- Condition-control loop: $\boxed{\text{while loop}}$ $\boxed{\text{DO-while loop}}$
 Repetition based on a condition

- i. Initialization
- ii. Termination
- iii. Condition

lec# 17 PF

03/6/2023

Q. Write a program for this sequence to
iterate 10 times,

16, 34, 51, 68

$16 + 0, (16 \times 2) + 2, (16 \times 3) + 3, (16 \times 4) + 4$

PF Lec #18:

6/10/23

Write a program to add or
subtract then ask if user
want to continue

PF lec#
23/10/23

schemas
buffer

character arrays are strings

\0 → delimiter

(in end of string)

float arr[];

char arr2[9] = {'M', 'u', 'z', 'a', 'm', 'm', 'l', 'l'};

char arr3[] = "Hello World";

char arr4[10] = "abcdefghijkl";

scanf("%s", &arr4);

Price and company name of markers
10 markers, print("This company is selling
cheapest")

int price[10];

char company[5]

Store info of 50 students

float arr[]
sub 1
sub 2
sub 3
sub 4
sub 5 } mid 01 and
mid 02 scores

[50][10]

print name of student who has highest total

PF lec#

24/10/23

Q. Enter two words, sort it alphabetically

Q. Given a string.
print encrypted version

Q. Write a program

use rand() fx

until $\text{rand()} \% 3 == 0$

printf name

Q. User input str[20]

perform encryption on 3 characters
location rand

(a < b) ? (c < b) ? printf(b) : ((b < a) ? (c < a) ? printf(c)
printf(c);

Find greatest character
of three

if (a > b)

if (c < b)

printf(b)

else if (c < a)

if (b < a)

printf(a)

else
printf(c)

often size of pointer = 2 bytes

14/11/23

Indirection (*) operator
(reference)

To refer to the
contents of the variable that
the pointer points to

Example:

```
int v=101;  
int *p=&v;
```

```
printf ("%d", *p); // prints value
```

Q. Traverse array by pointer by offset

Q. 3 array [10]

use pointer to access position
all values between

void pointer → only address

cannot
dereference

cannot perform pointer
arithmetic

24/11/23

Eg:

```
int *funcpointer(int a, char b, int c);
```

pointer to function

↳ same return type

assigning:

funcpointer = firstExample
or funcpointer = &firstExample

```
int firstExample() {
```

}

calling:

```
int answer = funcpointer(7, 'A', 2);
```