Structure Chart

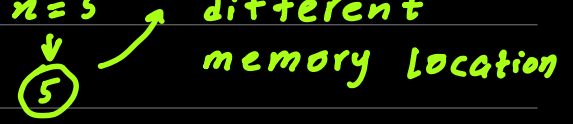
Modules : · Each sub-task is considered as a module.
· can be either a function / procedure

Parameters: Any value that has been passed b/w modules
· Input of a function

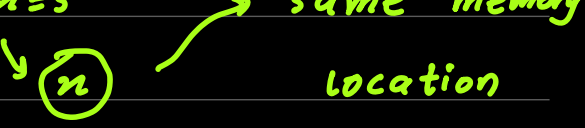
$f(x) = x^2 + 5$

- $f(1) = 1^2 + 5$
$= 6$

(1) **By Value :** Copy of variable is passed.
· Original value unchanged



(2) **By Reference:** Variable itself is passed.



· Original value changed.

- Structure Chart helps to understand the complex program and to identify the relation b/w modules

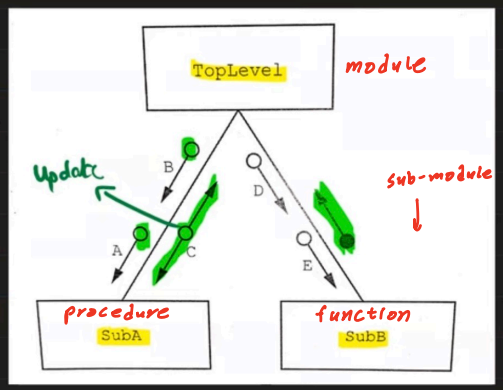
Header of a Function

Syntax : FUNCTION NAME (Name of variable : data type , Name of variable : Data type) RETURN Data type

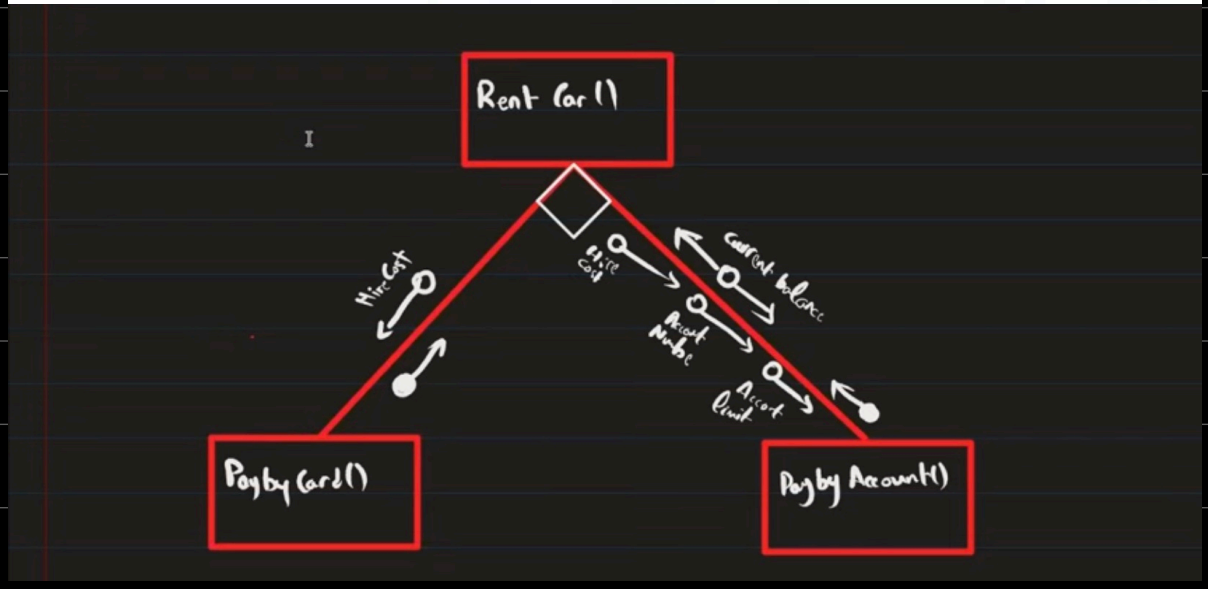
parameters

- The data type of the value obtained after performing the steps is to be written in "RETURN" Data type

SYMBOL	EXPLANATION
⟶ (STRING, INT, REAL)	value passed b/w modules
⟶	Boolean value passed b/w modules
◇	Selection statement
↻ ↺	Repitition
⟷	By reference (value changed) (update)



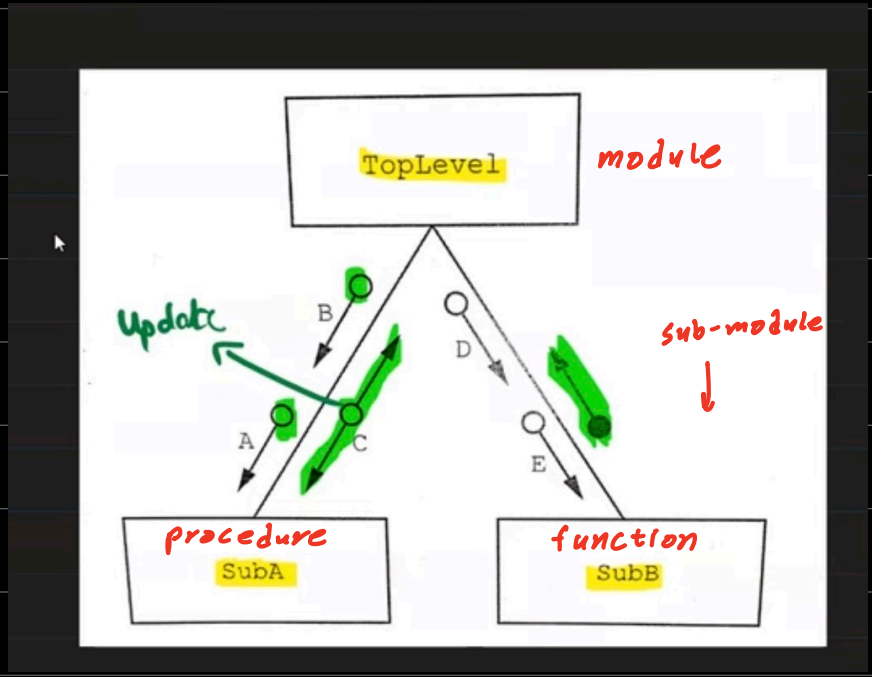
Module name	Description
`RentCar()`	A customer will pay for each car rental either by bank card or by using their account with the rental company.
`PayByCard()`	Called with parameter `HireCost`, representing the cost of the rental. Returns a `BOOLEAN` value to indicate whether or not the card payment was successful.
`PayByAccount()`	Called with parameters `HireCost`, `AccountNumber`, `CurrentBalance` and `AccountLimit`. • Checks whether `HireCost` plus the `CurrentBalance` would exceed the `AccountLimit`. If so, then the rental is not authorised. • If the rental is authorised, then the `CurrentBalance` is updated. • Returns a `BOOLEAN` value to indicate whether or not the rental was authorised.



A, D : string

C : Char

B, E : Integer



Q- Write pseudocode for module Sub B.

Function Sub B (O: string , E: Integer) Return Boolean

Q- Write pseudocode for module Sub A.

PROCEDURE (A: STRING, B: INTEGER , BYREF C : CHAR)

Features of Structure Chart

- Hierarchy of Modules
- Iteration
- Selection
- Parameters that are passed b/w modules
- Sequence of modules.