1 algorithm

Step 1 ⇒ Start

Step 2 ⇒ Read number of student in 'n'

Step 3 ⇒ Initialize $i = 0$

Step 4 ⇒ Repeat Steps 5 and 6 till $i <= n-1$

Step 5 ⇒ call a[i].getdata()

Step 6 ⇒ increment i by 1.

Step 7 ⇒ Initialize $i = 0$

Step 8 ⇒ Repeat Steps 9 and 10 till $i <= n-1$

Step 9 ⇒ Call a[i] display ()

Step 10 ⇒ Increment i by 1.

Step 11 ⇒ Declare const MAX = 100

Step 12 ⇒ Create a class basicinfo with data members name [20], rno and member functions getdata() and display()

Step 13 ⇒ Create a Subclass physicalfit that inherits the class basicinfo()

a) Data member - height, weight

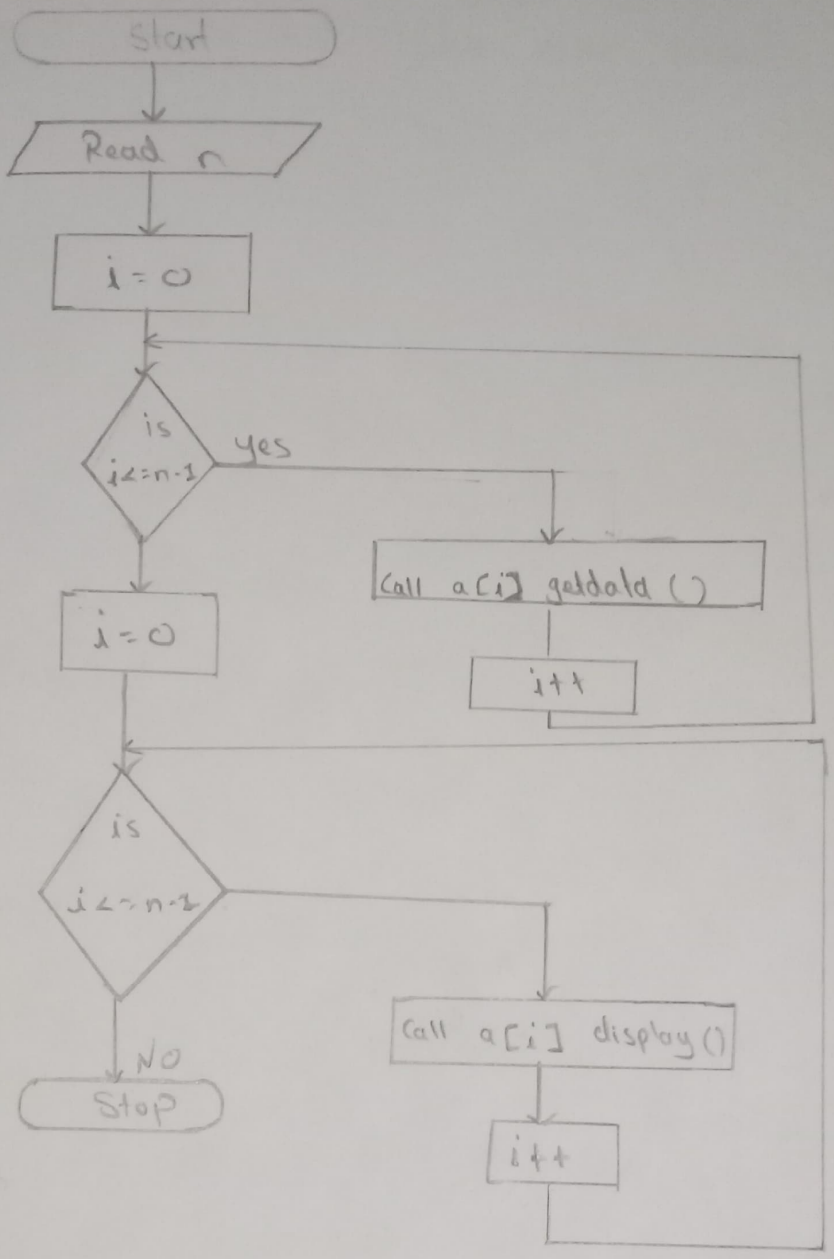
b) member functions - getdata(), display()

Step 14 ⇒ getdata() is used to take input for name and rno, height and weight.

Step 15 ⇒ display() is used for printing the name and rno, height and weight

Step 16 ⇒ Stop

2) Flowchart


Start
Read n
i = 0
is i<=n-1
yes
Call a[i] getdata ()
i++
i = 0
is i<=n-1
Call a[i] display ()
i++
NO
Stop

Coding

1 Algorithm :-2

Step 1:- Start

Step 2:- Declare class biggest with data member int a, b, c, large and member function getdata()

Step 3:- call getdata() to read a, b, c.

Step 4:- Declare friend function big()

Step 5:- Call big()

Step 6:- print largest number

Step 7:- Stop

# Algorithm:-3

Step 1:- Start

Step 2:- main method Start and Declare parameterzed in variable float.

Step 3:- Display passing function and two number massage

Step 4:- Read a and b

Step 5:- Display Screen addition Substration, and Read ch

Step 6:- if Statement and Condition (ch==a) operation Perform.

Step 7:- else part executed

Step 8:- massage's Show by one get one

Step 9:- using argument x, y

Step 10:- ans = x+y; and return (Ans)

Step 11:- Stop

# Algorithm:- 4

Step 1 :- Start

Step 2 :- Declare function and member variable

Step 3 :- Display massage and read n variable

Step 4 :- Operation performance temp = Sum(n)

Step 5 :- function calling in int Sum (int value=0;

Step 6 :- if Statement Create Condition (n=0) and return (value)

Step 7 :- elase part excute value = nt sun (n-1) Statement and return value

Step 8 :- Stope

# Algorithm 5

Step 1:- Start

Step 2:- Declare Data member variable

Step 3:- Display massage Read n

Step 4:- x = fact (n);

Step 5:- Show massage and x value factorial of x

Step 6:- function Call and argument pass

Step 7:- function implement and int value=1;

Step 8:- if (n==1) condition

Step 9:- using formula of factorial value = n * fact (n-1);

Step 10:- Stop

Writting

Algorithm:- 7

Step 1:- Start

Step 2:- declare Struct keyword

Step 3:- Public Declare function's and operator ++ void display ();

Step 4:- function implement fibonacci :: fibonccic()

Step 5:- by onemore function implement show fib.

Step 6:- operater function implement

Step 7:- Start main function and create objest fibonacci obj

Step 8:- using forloop and Statment

Step 9:- Stop

writting

Algorithm:- 8

Step 1:- Start

Step 2:- including headerfiles different

Step 3:- ~~Private~~ float class Declare triangle

Step 4:- Private float

Step 5:- call set. base function where b= base

Step 6:- call set. height function where h= height

Step 7:- Call getarea ();

Step 8:- Call operator = = ();

Step 9:- Print area.

Step 10:- Stop

writing

Algorithm 9:-

Step 1:- Start

Step 2:- include header file

Step 3:- main function start

Step 4:- using pointer variable

Step 5:- Initialize pointer
ptr -i , ptr_f, ptr_c , ptr_d.

Step 6:- Print ptr-i, ptr-f, ptr_c , ptr-d

Step 7:- delete ptr-i, ptr-f , ptr-c , ptr-d

Step 8:- Stop

Writting

Algorithm :- 10

Step 1:- Start

Step 2:- Class declare Student and member variable

Step 3:- public void getno and void putno ();

Step 4:- function implement with argument sno = a;

Step 5:- Display sno

Step 6:- Create Drive Class and Base class implement

Step 7:- Create variable float

Step 8:- void getmarks parameter and putmark function create

Step 9:- function implement test putmark()
Show sub1, sub2

Step 10:- Create inheritance Derive class bass. result and test

Step 11:- function call implement

Step 12:- Declare Derive class object

Step 13:- Create object

Step 14:- Stop