Algorithms for finding roots.

- 1) Bisection Method
 O Read a, b such that a < b & f(a)f(b) <

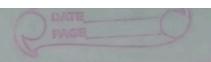
 m = (a+b)/2
- (3) If f(m) = 0 then m is an exact ecost, else if f(a). f(m) < 0 then b=m else if f(m), f(b) <0 then
- b=m

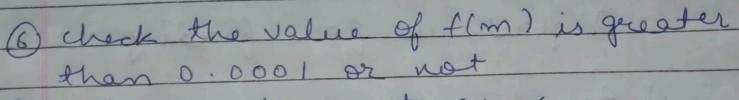
 (P) Repeat steps 2 & 3 until f(mi)=t

 (S) End else if

 (G) End

- 2) Regula Falsi Method-
- (1) Read values of 20, 20 & E = 0.00)
- 2) Function plue f(xo) & f(xi)
- 3) check whether product of f(xo) & t(x1) is negative or not.
- @ Determine m = xof(xi) xif(xo)
- f(x1) f(x0) 5) Check product x12 x0 is negative or not if (-ve) x0 = x, (+ve) x1 = x





a display the noot as x

® stop

3) Algorithm for Neluton's Method

@ Read b, E = 0.001

(3) 1=1 (4) while (1+lm) 1>E)

(3) print (i, b, m, f(m))

6) b < m

9 m = b - f(b) f'(b)

(8) i = i+1

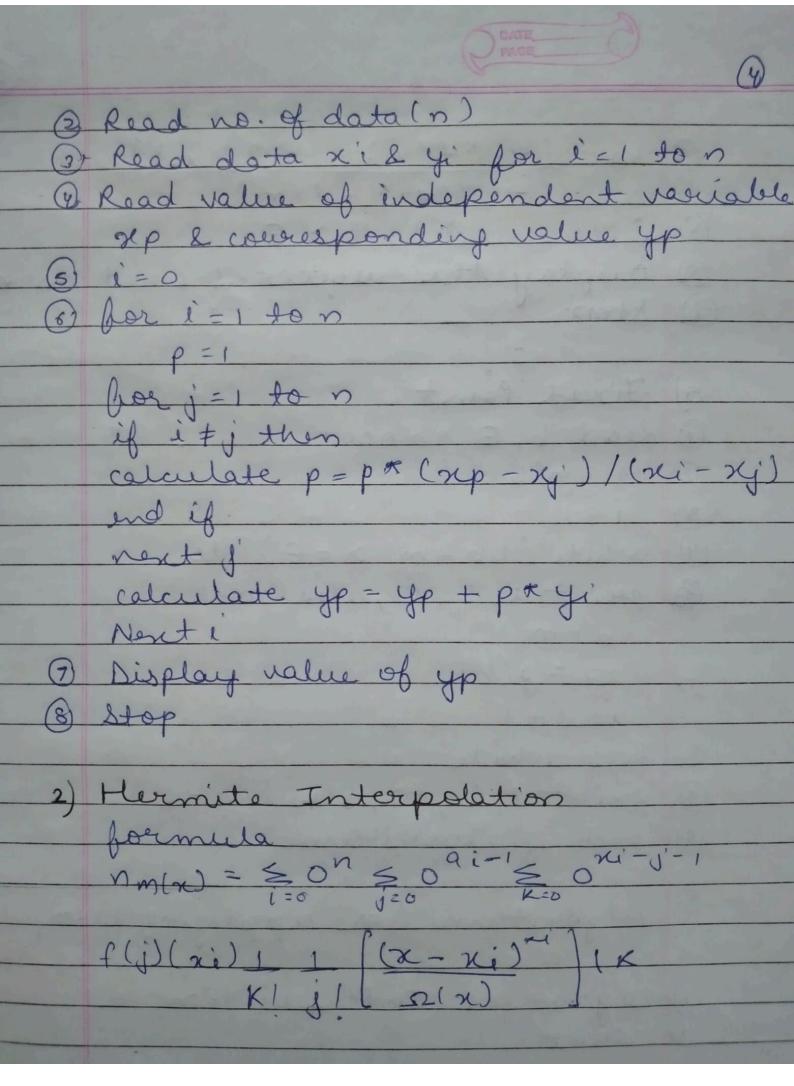
@ end of while

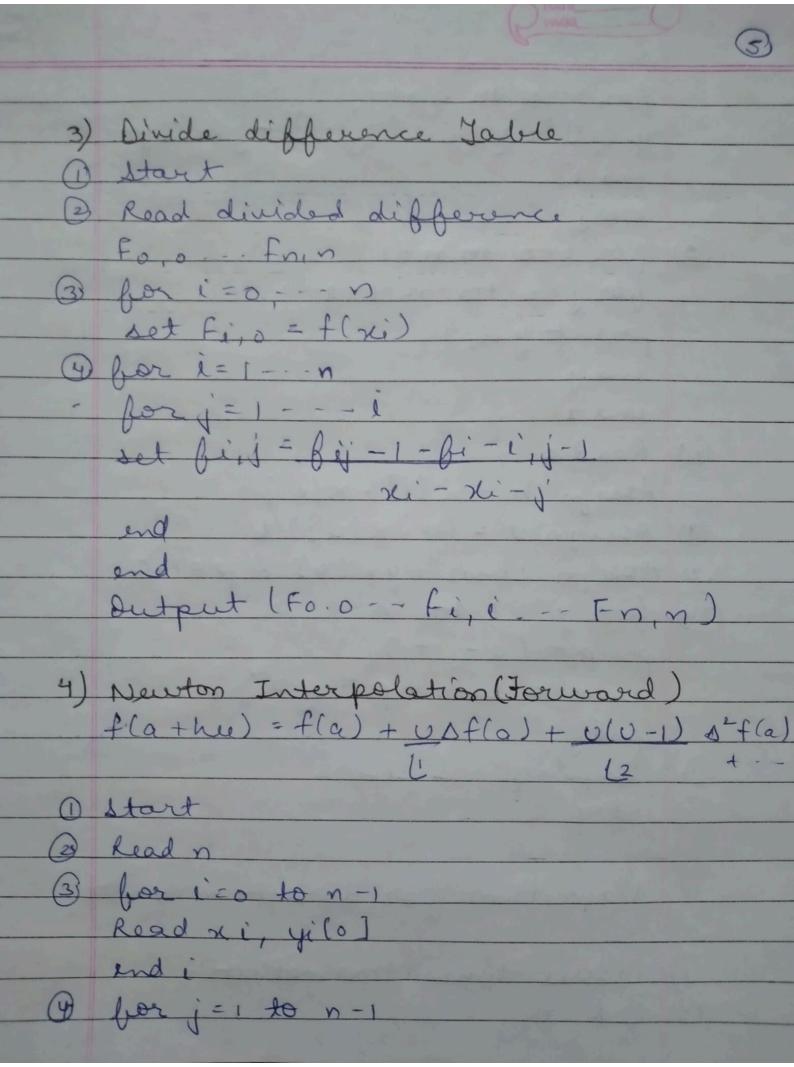
(10) end

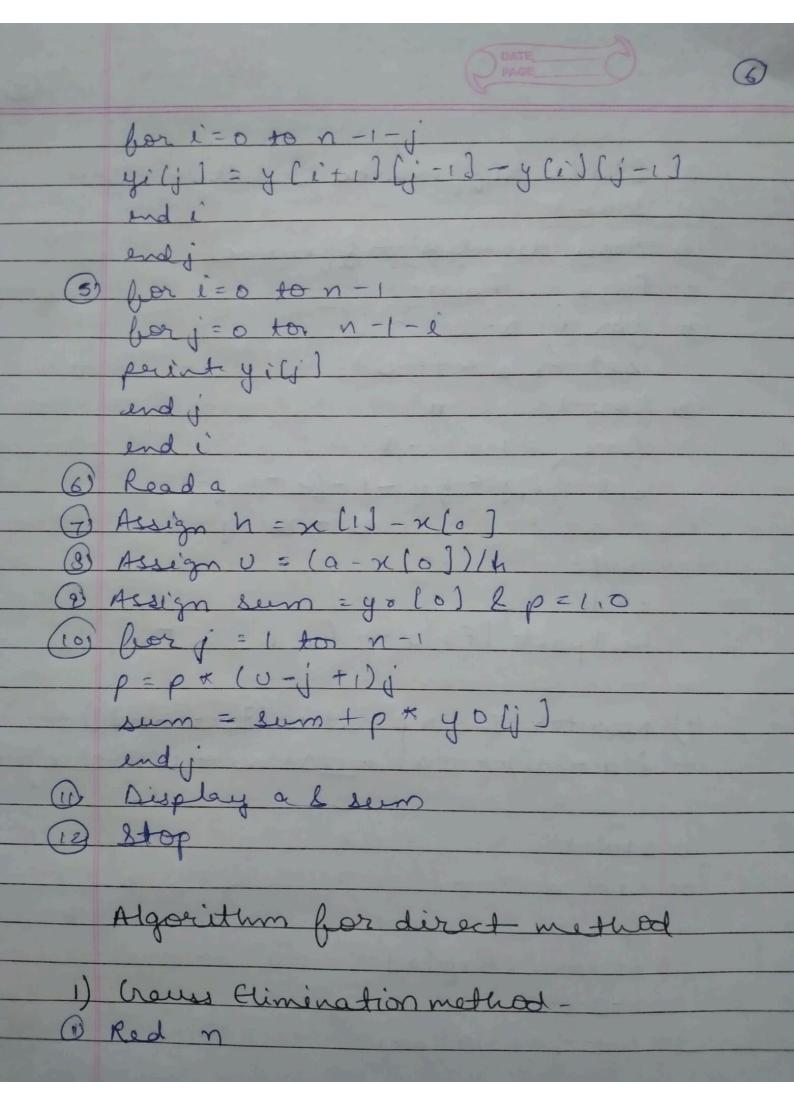
4) Secont Method

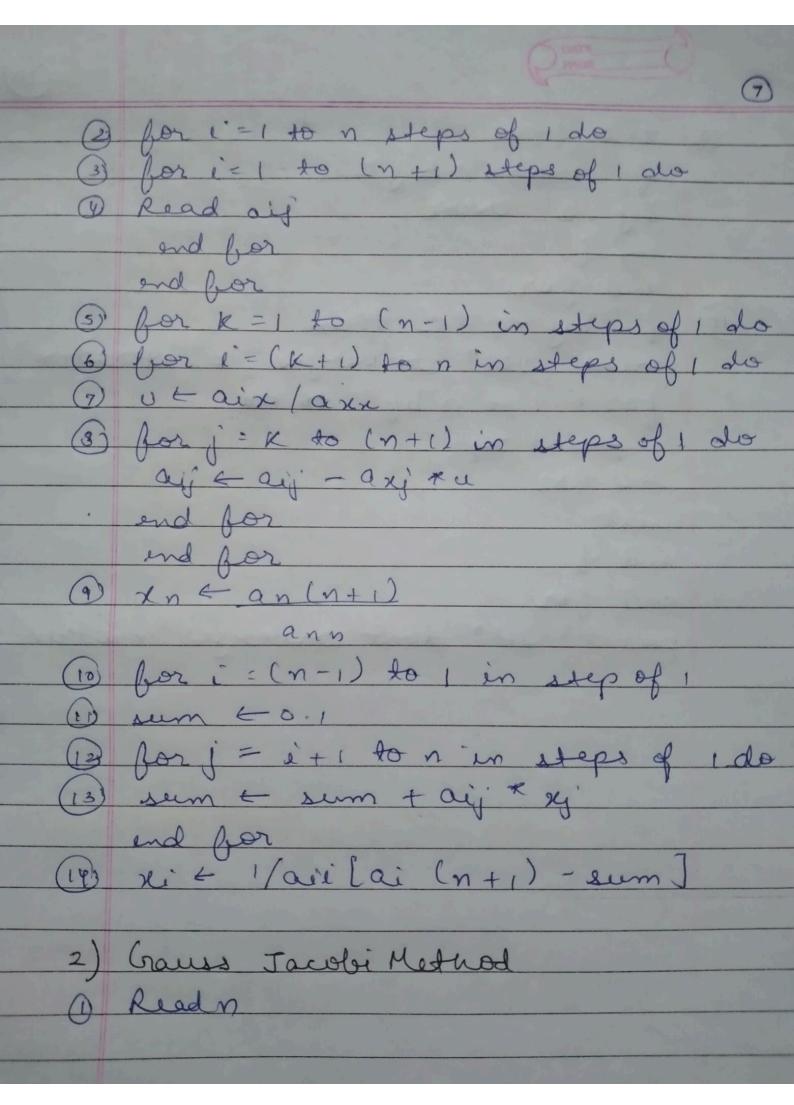
① Read value 20, 21, 8 € =0.001

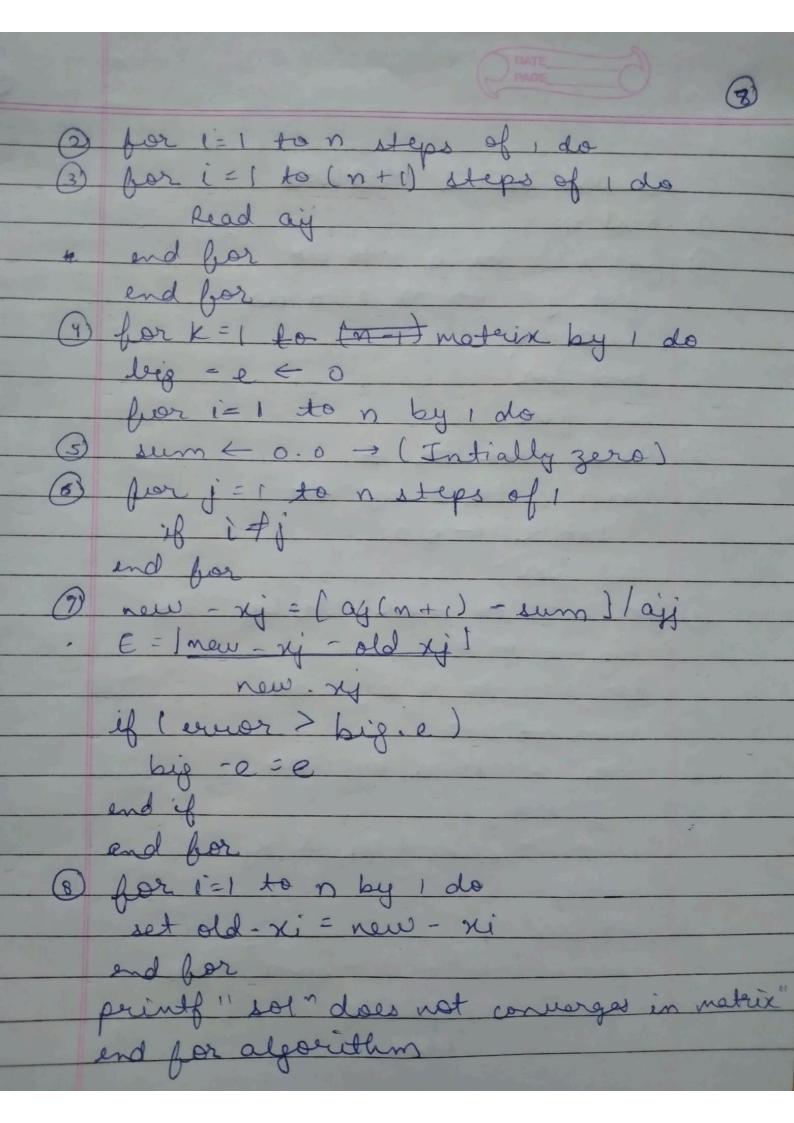
@ compute f(xo) & f(x1)











Crauss Saidel Method 1 Read on @ for i=1 to n 1 to do for j=1 to n 1 to do Read = oij end bor end for 3) Read max its E (y) for i=1 to n by I do bet x1 ← 0.0 end for (5) for K=1 to man, ite by 1 do big-e+0 for i=1 to n by 1 do sum < 0.0 6 hor j= 1 to n by 1 do if (i+j) sum t sum + aij x sej endif 3 temp + [ai(n+i) - sum] /aii error = temp - xi