

ฝึกปฏิบัติครั้งที่ 9

1. จงค้นคว้าตัวอย่างโปรแกรมการหาค่ารากสมการโดยกรรมวิธีการทำซ้ำแบบหนึ่งจุด
(One-point iteration Method)

```
1  """ สืบค้นจาก https://www.codesansar.com/numerical-methods/fixed-point-iteration-  
2  python-program-output.htm/ เมื่อ October 9,2021 """  
3  
4  # Fixed Point Iteration Method  
5  # Importing math to use sqrt function  
6  import math  
7  
8  def f(x):  
9      return x*x*x + x*x -1  
10  
11 # Re-writing f(x)=0 to x = g(x)  
12 def g(x):  
13     return 1/math.sqrt(1+x)  
14  
15 # Implementing Fixed Point Iteration Method  
16 def fixedPointIteration(x0, e, N):  
17     print('\n\n*** FIXED POINT ITERATION ***')  
18     step = 1  
19     flag = 1  
20     condition = True  
21     while condition:  
22         x1 = g(x0)  
23         print('Iteration-%d, x1 = %0.6f and f(x1) = %0.6f' % (step, x1, f(x1)))  
24         x0 = x1  
25  
26         step = step + 1  
27  
28         if step > N:  
29             flag=0  
30             break  
31  
32         condition = abs(f(x1)) > e  
33  
34     if flag==1:  
35         print('\nRequired root is: %0.8f' % x1)  
36     else:  
37         print('\nNot Convergent.')  
38  
39 # Input Section  
40 x0 = input('Enter Guess: ')  
41 e = input('Tolerable Error: ')  
42 N = input('Maximum Step: ')  
43  
44 # Converting x0 and e to float  
45 x0 = float(x0)  
46 e = float(e)  
47  
48 # Converting N to integer  
49 N = int(N)  
50  
51 # Starting Newton Raphson Method  
52 fixedPointIteration(x0,e,N)
```

2. จงอธิบายตัวอย่างโปรแกรมการหาคำรากสมการโดยกรรมวิธีการทำซ้ำแบบหนึ่งจุด (One-point iteration Method)

จากตัวอย่างโปรแกรม

- import module math เข้ามาใช้งานในการคำนวณ
- สร้างฟังก์ชัน f(x) : ในฟังก์ชันมีการ return ค่า จากการคำนวณของสมการ $x^3 + x^2 - 1$
- สร้างฟังก์ชัน g(x) : ที่ได้รับจาก $f(x)=0$ ดังนั้น $x = g(x)$ และ $|g'(x)| < 1$
- สร้างฟังก์ชัน fixedPointIteration() : มีพารามิเตอร์รับค่า 3 ตัว คือ การคาดเดาเบื้องต้น (x_0),

ข้อผิดพลาดที่ยอมรับได้ (e) และการวนซ้ำสูงสุด (N) เพื่อนำไปคำนวณภายในฟังก์ชัน

- วนลูปซ้ำ (while) จนกว่าเงื่อนไขจะเป็นเท็จ ก็คือ condition = false
 - คำนวณค่า $x_1 = g(x_0)$
 - กำหนดให้ค่า $x_0 = x_1$
 - เพิ่มตัวนับการวนลูปซ้ำ +1
 - เงื่อนไขถ้า ตัวนับการวนซ้ำ มากกว่า จำนวนการวนซ้ำสูงสุด (step > N) ให้กำหนด flag = 0
 - กำหนดให้ condition เป็นจริงหรือเท็จ เมื่อ $|f(x_1)| > e$ ถ้าเป็นจริง จะหยุดการวนซ้ำ
ถ้าเป็นเท็จ ให้วนลูปซ้ำต่อ
 - หาก condition เป็นเท็จแล้ว เข้าเงื่อนไข ถ้า flag = 1 ให้แสดงข้อความว่า “Required root is ...” หากไม่ใช่ ให้แสดงข้อความว่า “Not convergent”
- รับค่า x_0 , e และ N จากคีย์บอร์ด
 - เรียกใช้งานฟังก์ชัน fixedPointIteration() และส่งค่า argument ที่รับจากคีย์บอร์ด 3 ค่า คือ x_0 , e และ N

3. จงค้นคว้าตัวอย่างโปรแกรมการหาค่ารากสมการโดยกรรมวิธีนิวตัน-ราฟสัน(Newton-Raphson Method)

```
1  /* สืบค้นจาก https://www.geeksforgeeks.org/program-  
2  pfor-newton-raphson-method/ เมื่อ October 9,2021 */  
3  
4  // Java program for implementation of  
5  // Newton Raphson Method for solving  
6  // equations  
7  class GFG {  
8  
9      static final double EPSILON = 0.001;  
10  
11     // An example function whose solution  
12     // is determined using Bisection Method.  
13     // The function is  $x^3 - x^2 + 2$   
14     static double func(double x){  
15         return x * x * x - x * x + 2;  
16     }  
17  
18     // Derivative of the above function  
19     // which is  $3*x^2 - 2*x$   
20     static double derivFunc(double x){  
21         return 3 * x * x - 2 * x;  
22     }  
23  
24     // Function to find the root  
25     static void newtonRaphson(double x){  
26         double h = func(x) / derivFunc(x);  
27         while (Math.abs(h) >= EPSILON){  
28             h = func(x) / derivFunc(x);  
29  
30             //  $x(i+1) = x(i) - f(x) / f'(x)$   
31             x = x - h;  
32         }  
33  
34         System.out.print("The value of the"  
35             + " root is : "  
36             + Math.round(x * 100.0) / 100.0);  
37     }  
38     public static void main (String[] args){  
39  
40         // Initial values assumed  
41         double x0 = -20;  
42         newtonRaphson(x0);  
43     }  
44 }
```

4. จงอธิบายตัวอย่างโปรแกรมการหาค่ารากสมการโดยกรรมวิธีนิวตัน-ราฟสัน(Newton-Raphson Method)

จากตัวอย่างโปรแกรม

- สร้างคลาสชื่อ GFG
- สร้างตัวแปร 1 ตัวชื่อ EPSILON เป็นตัวแปรแบบ static และเป็นตัวแปรค่าคงที่มีชนิดข้อมูลเป็น double และกำหนดค่าให้เท่ากับ 0.1
- ภายในคลาส GFG มีเมธอดอยู่ 4 เมธอด ได้แก่
 1. เมธอด func() : เป็นเมธอดแบบ static มีการ return ค่าเป็นชนิดข้อมูลแบบ double และมีพารามิเตอร์ในการรับค่ามาคำนวณในเมธอดนี้ 1 ตัว คือ x รับค่าข้อมูลชนิดเป็น double , ภายในเมธอดมีการคำนวณตัวเลขจากสมการ $x^3 - x^2 + 2$ และส่งค่าผลลัพธ์จากการคำนวณคืนกลับไปยังที่ที่เมธอดถูกเรียกใช้
 2. เมธอด derivFunc() : เป็นเมธอดการหาอนุพันธ์ของสมการ $x^3 - x^2 + 2$ จะได้อนุพันธ์ $3x^2 - 2x$
 3. เมธอด newtonRaphson() : เป็นเมธอดการหาค่ารากสมการโดยกรรมวิธีนิวตัน-ราฟสัน โดยมีลำดับขั้นตอนดังนี้
 - คำนวณค่าของ func(x) และ derivFunc(x) สำหรับ x . เริ่มต้นที่กำหนด
 - คำนวณ h: $h = \text{func}(x) / \text{derivFunc}(x)$
 - ถ้าค่าของ h มากกว่า ค่าคลาดเคลื่อนที่ user ยอมรับได้ (Epsilon)
ให้คำนวณ $h = \text{func}(x) / \text{derivFunc}(x)$
และคำนวณ $x = x - h$ // $(x(i+1) = x(i) - f(x) / f'(x))$
 4. เมธอด main() : เป็นเมธอดที่เป็นส่วนแสดงผลของโปรแกรม โดยในเมธอดทำการสร้างตัวแปร x0 มีชนิดข้อมูลเป็น double กำหนดค่า = -20 และทำการเรียกใช้งานเมธอด newtonRaphson() และส่งค่า Argument 1ค่า คือ ส่งค่า x0 ไปเพื่อทำการคำนวณหาค่ารากสมการในเมธอดนั้น

5. จงค้นคว้าตัวอย่างโปรแกรมการหาค่ารากสมการโดยกรรมวิธีเซแคนต์ (Secant Method)

```
1  /* สืบค้นจาก https://www.geeksforgeeks.org/program-to-find-root-  
2  of-an-equations-using-secant-method/ เมื่อ October 9,2021 */  
3  
4  // Java Program to find root of an  
5  // equations using secant method  
6  class GFG {  
7  
8      // function takes value of x and  
9      // returns f(x)  
10     static float f(float x) {  
11         // we are taking equation  
12         // as x^3+x-1  
13         float f = (float)Math.pow(x, 3) + x - 1;  
14         return f;  
15     }  
16  
17     static void secant(float x1, float x2, float E) {  
18  
19         float n = 0, xm, x0, c;  
20         if (f(x1) * f(x2) < 0){  
21             do {  
22  
23                 // calculate the intermediate  
24                 // value  
25                 x0 = (x1 * f(x2) - x2 * f(x1)) / (f(x2) - f(x1));  
26  
27                 // check if x0 is root of  
28                 // equation or not  
29                 c = f(x1) * f(x0);  
30  
31                 // update the value of interval  
32                 x1 = x2;  
33                 x2 = x0;  
34  
35                 // update number of iteration  
36                 n++;  
37  
38                 // if x0 is the root of equation  
39                 // then break the loop  
40                 if (c == 0)  
41                     break;  
42                 xm = (x1 * f(x2) - x2 * f(x1)) / (f(x2) - f(x1));  
43  
44                 // repeat the loop until the  
45                 // convergence  
46             } while (Math.abs(xm - x0) >= E);  
47  
48             System.out.println("Root of the" + " given equation=" + x0);  
49  
50             System.out.println("No. of " + "iterations = " + n);  
51         }  
52  
53         else  
54             System.out.print("Can not find a"  
55             + " root in the given interval");  
56     }  
57     public static void main(String[] args) {  
58  
59         // initializing the values  
60         float x1 = 0, x2 = 1, E = 0.0001f;  
61         secant(x1, x2, E);  
62     }  
63 }
```

6. จงอธิบายตัวอย่างโปรแกรมการหาค่ารากสมการโดยกรรมวิธีเซแคนต์ (Secant Method)

จากตัวอย่างโปรแกรม

- สร้างคลาสชื่อ GFG
- ภายในคลาส GFG มีเมธอดอยู่ 3 เมธอด ได้แก่
 1. เมธอด f() : เป็นตัวแปรแบบ static มีการ return ค่าชนิดข้อมูลเป็น float และมีพารามิเตอร์ในการรับค่า 1 ตัว คือ x รับค่าข้อมูลชนิดเป็น double , ภายในสร้างตัวแปร f มีชนิดข้อมูลเป็น float เก็บค่าจากการคำนวณของสมการ เมธอดมีการคำนวณตัวเลขจากสมการ $x^3 + x - 1$ และส่งค่าจากตัวแปร f กลับไปยังที่ที่เมธอดถูกเรียกใช้
 2. เมธอด secant() : เป็นตัวแปรแบบ static ไม่มีการ return ค่า และมีพารามิเตอร์ในการรับค่า 3 ตัว คือ x1, x2 และ E เพื่อนำคำนวณภายในเมธอด secant() และสร้างตัวแปรชนิดข้อมูลเป็น float ได้แก่ n กำหนดค่า=0 , xm , x0 และ c จากนั้นเข้าเงื่อนไข ถ้าเงื่อนไขเป็นจริง ถ้า $f(x1) * f(x2) < 0$ ให้ไปทำลูปวนซ้ำ (do-while) โดยทำงานที่ลูป do ก่อน 1 รอบเสมอ ก่อนตรวจสอบเงื่อนไขที่กำหนด และจะทำซ้ำการทำงานไปเรื่อย ๆ จนกว่าเงื่อนไขในลูป while จะเป็นเท็จ เมื่อ $|(xm - x0)| \geq E$ และแสดงข้อความพร้อมค่าของราก (**x0**) และจำนวนการวนซ้ำ (**n**) และ ถ้าเงื่อนไขเป็นเท็จ ให้แสดงข้อความว่า "ไม่พบรากในช่วงเวลาที่กำหนด"
 3. เมธอด main() : เป็นเมธอดที่เป็นส่วนแสดงผลของโปรแกรม โดยในเมธอดทำการสร้างตัวแปรชนิดข้อมูลเป็น float ได้แก่ x1 กำหนดค่า = 0 , x2 กำหนดค่า = 2 , E = 0.0001 และทำการเรียกใช้งานเมธอด secant() และส่งค่า Argument 3 ค่า คือ ส่งค่า x1, x2 และ E ไปเพื่อทำการคำนวณหาค่ารากสมการในเมธอดนั้น