Assignment 1 110550085 [illegible]

1. (a). Bisection Method:

```
double f(double x){
    return x*x + sin(x)-(exp(x)/4)-1;
}

int main(){
    double a,b;
    cin a>>b;
    while (abs(b-a)<0.00001){
        double m=(a+b)/2;
        if(f(a)*f(m)<0){
            b=m;
        }
        else{
            a=m;
        }
    cout<<a;
    return 0;
}
```

Ans: -1.43181, 0.911911.

(b) Secant Method:

```
double f(double x){
    return x*x+sin(x)-(exp(x)/4)-1);
}
int main(){
    double a,b;
    cin>>a>>b;
    while(abs(b-a)>0.00001){
        double m=b-(f(b)*(a-b)/(f(a)-f(b)));
        if(abs(f(a))<abs(f(b))){
            b=a;
            a=m;
        }
        else{
            a=b;
            b=m;
        }
    }
    cout<<a;
    return 0;
}
```

Ans: -1.43181, 0.911918.

1.(c). Newton's Method,

$f'(x) = 2x + \cos(x) - \frac{e^x}{4}$.

$\Rightarrow$

```
double f(double x){
    return x*x+sin(x)-(exp(x)/4)-1;
}
double df(double x){
    return 2*x+cos(x)-(exp(x)/4);
}
int main(){
    double a,b;
    cin>>a>>b;
    while(abs(a-b)>0.00001){
        a=b;
        b=a-f(a)/df(a);
    }
    cout<<a;
}
```

Ans: <u>0.911918 or -1.43181</u>

(Both $[-2,0]$ and $[0,2]$ yield the result 0.911918, I chose $[-2,-1]$ to get the result of -1.43181.)

2. $P(x) = (x-2)^3(x-4)^2$

$P'(x) = 3(x-2)^2(x-4)^2 + 2(x-2)^3(x-4)$

$P'(x_0) = \frac{P(x_0)}{(x_0 - x_1)}$.

$x_1 = x_0 - \frac{P(x_0)}{P'(x_0)}$

$\Rightarrow P(3) = 1,\ P'(3) = 3 - 2 = 1$

$x_1 = 3 - \frac{1}{1} = 2$, so <u>$P(x)$ converges to 2</u>.

$P'(2) = 0$.

Then, $\lim_{x \to 2} \frac{|2 - (x - \frac{P(x)}{P'(x)})|}{|2 - x|} = \lim_{x \to 2^-} \frac{2 - x + \frac{(x-2)^3(x-4)^2}{3(x-2)^2(x-4)^2 + 2(x-2)^3(x-4)}}{2 - x}$

$= \lim_{x \to 2^-} \frac{2 - x + \frac{(x-2)(x-4)}{3(x-4) + 2(x-2)}}{2 - x} = \lim_{x \to 2^-} 1 - \frac{x-4}{5x-16} = 1 - \frac{-2}{-6} = \frac{2}{3} > 0$,

So it converges linearly.

$\Rightarrow$ (a) Yes.

$\Rightarrow$ (b) 2.

(c) No, the convergence is linear.

3. (a).

$$x-\frac{4+2x^3}{x^2}+2x=0$$
$$3x-\frac{4+2x^3}{x^2}=0$$
$$3x=\frac{4+2x^3}{x^2}$$
$$x^3=4$$
$$x^3-4=0$$

(b).

$$x-\sqrt{\frac{4}{x}}=0$$
$$x=\sqrt{\frac{4}{x}}$$
$$x^2=\frac{4}{x}$$
$$x^3=4.$$
$$x^3-4=0$$

(c)

$$x-\frac{16+x^3}{5x^2}=0$$
$$x=\frac{16+x^3}{5x^2}$$
$$5x^3=16+x^3$$
$$4x^3=16$$
$$x^3=4.$$
$$x^3-4=0$$

$\Rightarrow$ By (a)(b)(c), $\underline{f(x)=x^3-4.}$

(a)

$$g(x)=\frac{4+2x^3}{x^2}-2x=\frac{4}{x^2}.$$
$$g'(x)=\frac{-8x}{x^4}=-\frac{8}{x^3}.$$
$$|g'(R)|=\left|\frac{-8}{(\sqrt[3]{4})^3}\right|=2.\Rightarrow \text{It diverges.}$$

(b)

$$g(x)=\sqrt{\frac{4}{x}}=2x^{-\frac{1}{2}}$$
$$g'(x)=-x^{-\frac{3}{2}}$$
$$|g'(R)|=\left|\sqrt[3]{4}^{-\frac{3}{2}}\right|=\left|(4^{\frac{1}{3}})^{-\frac{3}{2}}\right|=\left|-4^{-\frac{1}{2}}\right|=\frac{1}{\sqrt{2}}.<1\Rightarrow \text{It converges.}$$

(c)

$$g(x)=\frac{16+x^3}{5x^2}=\frac{16}{5}x^{-2}+\frac{1}{5}x$$
$$g'(x)=-\frac{32}{5}x^{-3}+\frac{1}{5}$$
$$|g'(R)|=\left|-\frac{32}{5}\cdot 4^{-1}+\frac{1}{5}\right|=\left|-\frac{8}{5}+\frac{1}{5}\right|=\frac{7}{5}>1\Rightarrow \text{It diverges.}$$

$\Rightarrow$ (b) converges.

$\Rightarrow$ Run the following C++ code to obtain x:

```
double g(double x){
    return sqrt(4/x);
}
int main(){
  double a,b;
  cin>>a;
  b=g(a);
  while(abs(b-a)>0.00001){
      a=b;
      b=g(a);
  }
  cout<<a;
}
```

$\Rightarrow$ We get $\underline{x=1.5874}$.

$\Rightarrow$ So the Answers are here:

$\underline{f(x)=x^3-4}$, $\underline{\text{(b) converges}}$, $\underline{x\cong 1.5874}$, $\underline{\text{all } x_0 \text{ for (a)(c) diverge}}$. $\underline{\text{(a)(c) diverge.}}$

4. $\begin{cases} x-3y-z^2+3 \\ 2x^3+y-5z^2+2 \\ 4x^2+y+z-7 \end{cases}$

$$\Rightarrow J = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} & \frac{\partial f_1}{\partial z} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} & \frac{\partial f_2}{\partial z} \\ \frac{\partial f_3}{\partial x} & \frac{\partial f_3}{\partial y} & \frac{\partial f_3}{\partial z} \end{bmatrix} = \begin{bmatrix} 1 & -3 & -2z \\ 6x^2 & 1 & -10z \\ 8x & 1 & 1 \end{bmatrix}.$$

Use the following Python code to solve it:
(Also can be seen in the file I submit).

```
import numpy as np
def jacobian(xyz):
  x,y,z= xyz
  return [[1,-3,-2*z],
          [6*x*x,1,-10*z],
          [8*x,1,1]].
def f(x,y,z):
  x,y,z=xyz
  return [x-3*y-z*z+3, 2*x*x*x+y-5*z*z+2, 4*x*x+y+z-7]
def newton(f, x_init, jacobian):
   max_iter = 100
   tol= 0.00001
   x_last= x_init
   for i in range(max_iter):
      J= np.array(jacobian(x_last))
      F= np.array(f(x_last))
      diff= np.linalg.solve(J,-F)
      x_last= x_last + diff
      if(np.linalg.norm(diff)<tol):
          return x_last

   return x_last

a= complex(input())
b= complex(input())
c= complex(input())
x_sol= newton(f,[a,b,c], jacobian)
print(x_sol)
```

$\Rightarrow$ The solutions and the corresponding first guesses are the following:

Guess	Solution
(1,1,1)	(1.11140818, 0.98820972, 1.07087768)
(1.3,0.9,-1.2)	(1.35374829, 0.92543063, -1.28596832)
(-1,-1,-1)	(32.89463066, -4434.08662023, 115.49088488)
(30,-3,-7)	(31.15140483, -3768.15712611, -106.48296945)
(-2+4j,-3+20j,7+9j)	(-1.25059599+0.04899466j, 0.665053-0.01340966j, 0.0885876+0.50358986j)
(-2-4j,-3-20j,7-9j)	(-1.25059599-0.04899466j, 0.665053+0.01340966j, 0.0885876-0.50358986j).