```c
#include<stdio.h>
#include<math.h>


double eps(double e)
{
    return 1.0 + e / 2.0 > 1.0 ? eps(e / 2.0) : e;
}
double f(double x)
{
    return cos(x) - exp(- (x * x) / 2) + x - 1;
}
double fd(double x)
{
    return -sin(x) + exp(- (x * x) / 2) * x + 1;
}
double g(double x)
{
    return 1 - x + sin(x) -log(1 + x);
}
double gd(double x)
{
    return -1 + cos(x) -1 / (1 + x);
}
double iter(double a, double b)
{
    double e = eps(1.0);
    double x;
    double xP1 = (a + b) / 2;
    do {
        x = xP1;
        xP1 = f(x) + x;
    } while (fabs(xP1 - x) > e) ;
    return xP1;
}


double newt(double a, double b)
{
    double e = eps(1.0);
    double x;
```

```c
    double xP1 = (a + b) / 2;
    do {
        x = xP1;
        xP1 = x - f(x) / fd(x);
    } while (fabs(xP1 - x) > e) ;
    return xP1;
}

double dich(double a, double b)
{
    double e = eps(1.0);
    while ((b - a) > e * 10) {
        if (f(a) * f((a + b) / 2) < 0) {
            b = (a + b) / 2;
        } else if (f(a) * f((a + b) / 2) > 0) {
            a = (a + b) / 2;
        } else if (f(a) == 0) {
            b = a;
            break;
        } else if (f(b) == 0) {
            a = b;
            break;
        }
    }
    return (a + b) / 2;
}

int main(void){

    double a = 1.0;
    double b = 2;

    printf("Дихотомия      %.25lf\nИтерационный %.25lf\nНьютьон        %.25lf\n", dich(a, b), iter(a, b), newt(a,b));
    for (double x = a; x <= b; x += 0.01) {
        printf("%.25lf\n", fabs(fd(x) + 1));
    }

    return 0;
}
```