

```
#include<bits/stdc++.h>
```

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

```
using namespace std;
```

```
double func(double x)
```

```
{
```

```
    return 2*x*x*x + 3*x - 1;
```

```
}
```

```
double es=0.001;
```

```
double x0, r_error, abs_error, f0, f1, f2;
```

```
int i=0;
```

```
void bisection(double x_lower, double x_upper)
```

```
{
```

```
    f1=func(x_lower);
```

```
    f2=func(x_upper);
```

```
    if(f1*f2>0)
```

```
    {
```

```
        printf("No root lying between %lf and %lf", x_lower, x_upper);
```

```
        return;
```

```
    }
```

```
    r_error=fabs((x_upper-x_lower)/x_upper);
```

```
while(r_error>es)
```

```
{
```

```
    r_error=fabs((x_upper-x_lower)/x_upper);
```

```

f1=func(x_lower);
f2=func(x_upper);

x0=(x_lower + x_upper)/2;
f0=func(x0);

if(f1*f0<0)
{
    i++;
    x_upper=x0;
    abs_error= abs(x0-x_lower);
    r_error=fabs((x_upper-x_lower)/x_upper);
    printf("No of itr=%d\t x0=%lf\t Abs_Error=%lf\t Relative_Error=%lf\n", i, x0, abs_error,
r_error);
}
else{
    i++;
    x_lower=x0;
    abs_error= abs(x_upper-x0);
    r_error=fabs((x_upper-x_lower)/x_upper);
    printf("No of itr=%d\t x0=%lf\t Abs_Error=%lf\t Relative_Error=%lf\n", i, x0, abs_error,
r_error);
}
}

int main()
{

```

```
double x_lower, x_upper;

printf("Enter the value of x_lower & x_upper:");

printf("\n");

scanf("%lf %lf", &x_lower, &x_upper);


bisection(x_lower, x_upper);

printf("\n");

printf("Approximate root: %lf\n", x0);


return 0;

}
```