```cpp
#include <iostream>
#include <fstream>
#include <cmath>

using namespace std;




double K = 4000;
double r = 0.2;
double e = 2.71828;
double a = 0, b = 50;
double alpha = 1000;


double y_exact(double t) {

    double y_0 = 1000;

    return (y_0 * K)/(y_0 + (K - y_0)*pow(e, -r*t));

}


double y_prime(double y) {

    return r * y * ( 1 - (y/K));

}

//Input:  endpoints a, b, int N, init cond (alpha)
//Output: approx w to y at N+1 values of t

double Eulers(double h) {

    int N = (b-a)/h;
    double t = a;
    double w = alpha;


    double max_diff = 0;
    int t_diff;

    cout << "N: " << a <<  " t: " << t <<  "  w: " << w <<endl;

    for(int i = 1; i <= N; i++) {

        w = w + h * y_prime(w);
```

```cpp
    t = a + i * h;

    double exact = y_exact(t);

    if( exact - w > max_diff ) {
      max_diff = exact - w;
      t_diff = t;
    }

    cout << t << "      " << w << endl;

  }


  cout << endl << " max difference is: " << max_diff
       << " and occurs at t = " << t_diff << endl;

  return 0;

}




int main() {

  Eulers(10);
  cout <<endl<<endl;
  Eulers(1);
  cout <<endl<<endl;
  Eulers(.1);
  cout <<endl<<endl;


}
```