```
#include <iostream>
#include <chrono>
#include <vector>
#include <iomanip>
#include <cmath>
#include <string>
#pragma GCC optimize("O0")


using namespace std;







double maximum(vector<double> a){
    double tmp = a[0];
    for (int i = 1; i < a.size(); i++){
        if (tmp < a[i]){tmp = a[i];}
    }
    return tmp;
}



template<typename T>
T plusic(T a, T b){return a + b;}

template<typename T>
T minusic(T a, T b){return a - b;}

template<typename T>
T mult(T a, T b){return a * b;}

template<typename T>
T divid(T a, T b){return a / (!b + b);}

template<typename T>
T nothing(T a, T b){return a;}

template<typename T>
T divid_helper(T a, T b){return (!b + b);}


long long QUON_OF_ITER = pow(10, 6);


template<typename T>
double iter(T (*foo)(T, T), T a, T b, T k){
    auto c = foo(a, b);
    auto begin = std::chrono::steady_clock::now();
```

```
    for (long long i = 0; i < QUON_OF_ITER; i++){
        b = foo(a, b);
        a = foo(a, b);
        b = foo(a, b);
        a = foo(a, b);
        b = foo(a, b);
        a = foo(a, b);
        b = foo(a, b);
        a = foo(a, b);
        b = foo(a, b);
        a = foo(a, b);
    }

    auto end = std::chrono::steady_clock::now();

    auto dur = std::chrono::duration_cast<std::chrono::milliseconds>(end -
begin);

    return (double)dur.count() / 100000;
}


template<typename T>
double result(char c, T a, T b, T k){
    T (*i)(T, T);
         if (c == '+'){i = plusic;}
    else if (c == '-'){i = minusic;}
    else if (c == '*'){i = mult;}
    else              {i = divid;}
    return iter(i, a, b, k);
}



void output(vector<double>a){


    cout << "\n\n";

    cout << scientific;

    vector<string> types = {"int", "float", "double", "long double", "long
long", "char"};
    vector<string>probel = {"        ", "      ", "     ", "", "  ", "
"};
    vector<char> opers = {'+', '-', '*', '/'};
    double max = maximum(a);

    for (int i = 0; i < a.size(); i++){

        cout << setw(5) << "|" << opers[i % 4] << "|"
             << setw(9) << "|" << types[i / 4] << "|"
             << probel[i / 4]
```

```
            << setw(1) << "|" << a[i] << "|"
            << "  |" << string(int(25 * abs(a[i]) / max), 'X')
            << string(25 - int(25 * a[i] / max), ' ') << "|" << setw(10)
<< " | " << int(100 * a[i] / max) << "% |";


        cout << "\n";
        if (i % 4 == 3){cout << "\n";}
    }
}



int main(){


    vector<double> a, l, d;

    d.push_back(iter(divid_helper, (int)214, (int)2112, (int)10000));
    d.push_back(iter(divid_helper, (float)214.2413, (float)2112.2634,
(float)1000.42121));
    d.push_back(iter(divid_helper, (double)214.2413, (double)2112.2634,
(double)1000.42121));
    d.push_back(iter(divid_helper, (long double)21454326.32659, (long
double)21123423.2352135, (long double)10000000.3245214126));
    d.push_back(iter(divid_helper, (long long)2145432623, (long
long)2112342323, (long long)1000000000));
    d.push_back(iter(divid_helper, (char)'L', (char)'o', (char)'-'));

    l.push_back(iter(nothing, (int)214, (int)2112, (int)10000));
    l.push_back(iter(nothing, (float)214.2413, (float)2112.2634,
(float)1000.42121));
    l.push_back(iter(nothing, (double)214.2413, (double)2112.2634,
(double)1000.42121));
    l.push_back(iter(nothing, (long double)21454326.32659, (long
double)21123423.2352135, (long double)10000000.3245214126));
    l.push_back(iter(nothing, (long long)2145432623, (long
long)2112342323, (long long)1000000000));
    l.push_back(iter(nothing, (char)'L', (char)'o', (char)'-'));

    //int
    a.  push_back( QUON_OF_ITER / (result('+', (int)214, (int)2112,
        (int)10000) - l[0]));
    a.  push_back( QUON_OF_ITER / (result('-', (int)214, (int)2112,
        (int)10000) - l[0]));
    a.  push_back( QUON_OF_ITER / (result('*', (int)214, (int)2112,
        (int)10000) - l[0]));
    a.  push_back( QUON_OF_ITER / (result('/', (int)214, (int)2112,
        (int)10000) - d[0]));

    //float
    a.  push_back( QUON_OF_ITER / (result('+', (float)214.2413,
        (float)2112.2634, (float)1000.42121) - l[1]));
```

```
a.  push_back( QUON_OF_ITER / (result('-', (float)214.2413,
    (float)2112.2634, (float)1000.42121) - l[1]));
a.  push_back( QUON_OF_ITER / (result('*', (float)214.2413,
    (float)2112.2634, (float)1000.42121) - l[1]));
a.  push_back( QUON_OF_ITER / (result('/', (float)214.2413,
    (float)2112.2634, (float)1000.42121) - d[1]));

//double
a.  push_back( QUON_OF_ITER / (result('+', (double)214.2413,
    (double)2112.2634, (double)1000.42121) - l[2]));
a.  push_back( QUON_OF_ITER / (result('-', (double)214.2413,
    (double)2112.2634, (double)1000.42121) - l[2]));
a.  push_back( QUON_OF_ITER / (result('*', (double)214.2413,
    (double)2112.2634, (double)1000.42121) - l[2]));
a.  push_back( QUON_OF_ITER / (result('/', (double)214.2413,
    (double)2112.2634, (double)1000.42121) - d[2]));

//long double
a.  push_back( QUON_OF_ITER / (result('+', (long
    double)21454326.32659, (long double)21123423.2352135, (long
    double)10000000.3245214126) - l[3]));
a.  push_back( QUON_OF_ITER / (result('-', (long
    double)21454326.32659, (long double)21123423.2352135, (long
    double)10000000.3245214126) - l[3]));
a.  push_back( QUON_OF_ITER / (result('*', (long
    double)21454326.32659, (long double)21123423.2352135, (long
    double)10000000.3245214126) - l[3]));
a.  push_back( QUON_OF_ITER / (result('/', (long
    double)21454326.32659, (long double)21123423.2352135, (long
    double)10000000.3245214126) - d[3]));

//long long
a.  push_back( QUON_OF_ITER / (result('+', (long long)2145432623,
    (long long)2112342323, (long long)1000000000) - l[4]));
a.  push_back( QUON_OF_ITER / (result('-', (long long)2145432623,
    (long long)2112342323, (long long)1000000000) - l[4]));
a.  push_back( QUON_OF_ITER / (result('*', (long long)2145432623,
    (long long)2112342323, (long long)1000000000) - l[4]));
a.  push_back( QUON_OF_ITER / (result('/', (long long)2145432623,
    (long long)2112342323, (long long)1000000000) - d[4]));

//char
a.  push_back( QUON_OF_ITER / (result('+', (char)'L', (char)'o',
    (char)'-') - l[5]));
a.  push_back( QUON_OF_ITER / (result('-', (char)'L', (char)'o',
    (char)'-') - l[5]));
a.  push_back( QUON_OF_ITER / (result('*', (char)'L', (char)'o',
    (char)'-') - l[5]));
a.  push_back( QUON_OF_ITER / (result('/', (char)'L', (char)'o',
    (char)'-') - d[5]));

output(a);
```

```
    return 0;
}
```