

სკალარული ნამრავლი ბაიშვირიათებულ ვექტორში

სტანდარტული ალგორითმის ოპტიმიზაცია

```
cout<<"Computing scalar multiplication using optimized map algorithm\n";  
{  
    for(int i=0; i<N; i++){  
        map<int, double> x, y;  
        int n;  
        for(int j=0; j<N; j++){  
            x[j]=1.0;  
            y[j]=1.0;  
        }  
        start = clock();  
        double sum(0.0);  
        map<int, double>::iterator ix(x.begin()), iy(y.begin());  
        while(ix != x.end() && iy != y.end())  
        {  
            if( ix->first == iy->first )  
            {  
                sum+=ix->second * iy->second;  
                ++ix, ++iy;  
            }  
            else if( ix->first < iy->first ) ++ix;  
            else ++iy;  
        }  
        end = clock();  
        time[i]=((double)(end-start)/CLOCKS_PER_SEC);  
    }  
    cout<<"Computing scalar multiplication using optimized map algorithm\n";  
}
```

ხელმძღვანელი: კობა გელაშვილი
ავტორები: გიორგი გაჩეჩილაძე
რატი დევიძე
გიორგი ლულუნიშვილი

სიჩქარეთა შეფასება

[3% არანულოვანი ელემენტი]

