source code and test case on github

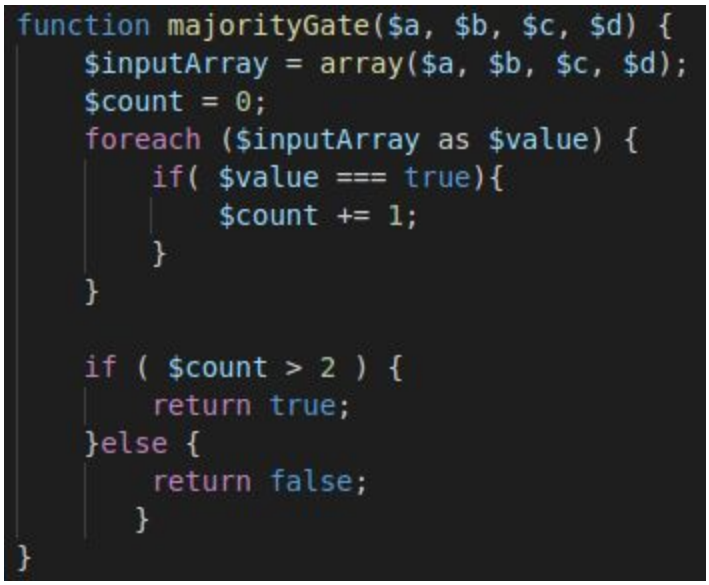
git clone https://github.com/kevinxzl/updatepromise.git

Build for C++:
g++ -std=c++11 updatepromise.cpp -o updatepromise

Run for GO:
go run updatepromise.go

1 majorityGate

PHP



```php
function majorityGate($a, $b, $c, $d) {
    $inputArray = array($a, $b, $c, $d);
    $count = 0;
    foreach ($inputArray as $value) {
        if( $value === true){
            $count += 1;
        }
    }

    if ( $count > 2 ) {
        return true;
    }else {
        return false;
       }
}
```

Go

```go
func majorityGate(a bool, b bool, c bool, d bool) bool {
    count := 0
    inputArr := [...]bool{a, b, c, d}
    for _, v := range inputArr {
        if v {
            count++
        }
    }
    if count > 2 {
        return true
    } else {
        return false
    }
}
```

C++

```cpp
bool majorityGate(bool a, bool b, bool c, bool d) {
    int count = 0;
    vector<bool> vt;
    vt.push_back(a);
    vt.push_back(b);
    vt.push_back(c);
    vt.push_back(d);
    for( auto it = vt.begin(); it != vt.end(); ++it) {
        if( *it ) {
            count++;
        }
    }

    if( count > 2 ) {
        return true;
    }else {
        return false;
    }
}
```

2 firstDuplicate

C++

```cpp
int firstDuplicate(vector<int> *vt) {
    set<int> mSet;
    pair<set<int>::iterator, bool> p;
    int index = 0;
    for( auto it = vt->begin(); it != vt->end(); ++it, ++index) {
        p = mSet.insert(*it);
        if( !(p.second) )  {
            return index;
        }
    }
    return -1;
}
```

Go

```go
func firstDuplicate(nums [] int) int {
    m := make(map[int]bool)
    for k, v := range nums {
        if _, flag := m[v]; !flag {
            m[v] = true
        }else {
            return k
        }
    }
    return -1
}
```

PHP

```php
function firstDuplicate($arr) {
    $len = count($arr);
    for( $i = 0; $i < $len-1; $i++) {
        for($j = $i+1; $j < $len; $j++) {
            if( $arr[$i] == $arr[$j] ) {
                return $j;
            }
        }
    }
    return -1;
}
```

3 frenchWeeks

C++

```cpp
float frenchWeeks( float weeks){
    if( weeks <= 0) {
        return 0;
    }
    float fweeks = 0.0;
    float rate = (float)(7 * 24 * 60 * 60) / (float)(10 * 10 * 100 * 100);
    return (weeks * rate);
}
```

GO

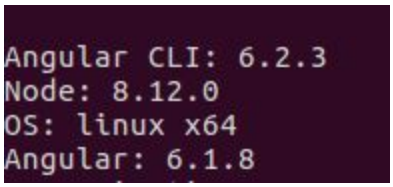
```go
func frenchWeeks(weeks float32) float32 {
    if weeks <= 0 {
        return 0
    }

    rate := float32( 7 * 24 * 60 * 60 ) / float32(10 * 10 * 100 * 100)
    return (weeks * rate)
}
```

PHP

```php
function frenchWeeks($weeks) {
    if( $weeks <= 0 ) {
        return 0;
    }
    $rate = (7 * 24 * 60 * 60) / (10 * 10 * 100 * 100);
    return ($rate * $weeks);
}
```

4 Angular Modular Concepts

1 install and run

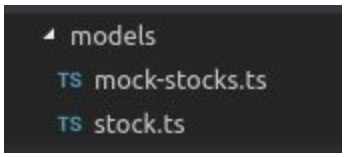


git clone https://github.com/kevinxzl/AngularModularConcepts.git
npm install
ng serve --open

2 create component

ng g c first
ng g c second
ng g c third

3 Create data model and test data for server




```
export class Stock {
    id: number;
    name: string;
    price: number;
}
```

```
export const STOCKS: Stock[] = [
    {id: 10000, name: 'AMD', price:31.02 },
    {id: 10001, name: 'MU' , price: 44.74 },
    {id: 10002, name: 'GE' , price:12.17 },
    {id: 10003, name: 'AAPL', price: 217.66},
    {id: 10004, name: 'SNAP', price: 9.14}
];
```

4 create service

ng g s service/stock

```
@Injectable({
  providedIn: 'root'
})
export class StockService {

  constructor() { }

  getStocks() : Observable <Stock[]> {
    return of(STOCKS);
  }

  getStock(name: string) : Observable<Stock> {
    return of(STOCKS.find(stock => stock.name === name))
  }


}
```

5 route

```
const routes: Routes = [
  { path: '', redirectTo: '/home', pathMatch: 'full' },
  { path: 'home', component: HomeComponent },
  { path: 'firstcomp', component: FirstComponent, children: [
    { path: 'thirdcomp/:id', component: ThirdComponent}
  ]},
  { path: 'secondcomp/:id', component: SecondComponent, children: [
    { path: 'thirdcomp/:id', component: ThirdComponent}
  ]},
  { path: '**', component: Code404Component }
];
```

```
    <div class="col-md-1">
      <a [routerLink]="['/']"> Home</a>
    </div>

    <div class="col-md-1">
        <a [routerLink]="['/firstcomp']" [queryParams]="{id:1}"> First Component </a>
    </div>

    <div class="col-md-1">
        <a [routerLink]="['/secondcomp', 2]"> Second Component </a>
    </div>

  </div>

<router-outlet></router-outlet>
</div>
```

6 UI


Home
First Component
Second Component
This is Second Component!
ID : 2
My Stocks
AMD price: 31.02
MU price: 44.74
GE price: 12.17
AAPL price: 217.66
SNAP price: 9.14
Third Component
This is Third Component!
Parent ID: 2