Java

```java
class TopVotedCandidate {

    public TopVotedCandidate(int[] persons, int[] times) {

    }

    public int q(int t) {

    }
}

/**
 * Your TopVotedCandidate object will be instantiated and called as such:
 * TopVotedCandidate obj = new TopVotedCandidate(persons, times);
 * int param_1 = obj.q(t);
 */
```

JavaScript

```javascript
/**
 * @param {number[]} persons
 * @param {number[]} times
 */
var TopVotedCandidate = function(persons, times) {

};

/**
 * @param {number} t
```

```
 * @return {number}
 */
TopVotedCandidate.prototype.q = function(t) {

};

/**
 * Your TopVotedCandidate object will be instantiated and called as such:
 * var obj = new TopVotedCandidate(persons, times)
 * var param_1 = obj.q(t)
 */
```

TypeScript

```
class TopVotedCandidate {
    constructor(persons: number[], times: number[]) {

    }

    q(t: number): number {

    }
}

/**
 * Your TopVotedCandidate object will be instantiated and called as such:
 * var obj = new TopVotedCandidate(persons, times)
 * var param_1 = obj.q(t)
 */
```

C++

```
class TopVotedCandidate {
public:
    TopVotedCandidate(vector<int>& persons, vector<int>& times) {

    }

    int q(int t) {

    }
};

/**
 * Your TopVotedCandidate object will be instantiated and called as such:
 * TopVotedCandidate* obj = new TopVotedCandidate(persons, times);
 * int param_1 = obj->q(t);
 */
```

C#

```
public class TopVotedCandidate {

    public TopVotedCandidate(int[] persons, int[] times) {

    }

    public int Q(int t) {

    }
```

```
}

/**
 * Your TopVotedCandidate object will be instantiated and called as such:
 * TopVotedCandidate obj = new TopVotedCandidate(persons, times);
 * int param_1 = obj.Q(t);
 */
```

Kotlin

```
class TopVotedCandidate(persons: IntArray, times: IntArray) {

    fun q(t: Int): Int {

    }

}

/**
 * Your TopVotedCandidate object will be instantiated and called as such:
 * var obj = TopVotedCandidate(persons, times)
 * var param_1 = obj.q(t)
 */
```

Go

```
type TopVotedCandidate struct {

}
```

```go
func Constructor(persons []int, times []int) TopVotedCandidate {

}


func (this *TopVotedCandidate) Q(t int) int {

}


/**
 * Your TopVotedCandidate object will be instantiated and called as such:
 * obj := Constructor(persons, times);
 * param_1 := obj.Q(t);
 */
```
