



Which of the following correctly declares a method that returns an one dimensional array?

- A. `public double[] computeAverage() { ... }`
- B. `public double[10] computeAverage() { ... }`
- C. `public double computeAverage() { ... }`
- D. `public double*10 computeAverage() { ... }`
- E. `public double[][] computeAverage() { ... }`

Understanding Arrays



cs6394q, Peer 6- Java Arrays - Fall Semester 2016

Question -3





Which of the following correctly declares a method that returns an one dimensional array?

- A. **`public double[] computeAverage() { ... }`**
- B. `public double[10] computeAverage() { ... }`
- C. `public double computeAverage() { ... }`
- D. `public double*10 computeAverage() { ... }`
- E. `public double[][] computeAverage() { ... }`

Understanding Arrays



cs6394q, Peer 6- Java Arrays - Fall Semester 2016

Question -4





Which invocation correctly passes the character array *cArray* to the method?

- A. `myMethod(cArray[]);`
- B. `myMethod(cArray);`
- C. `myMethod(cArray[8]);`
- D. `myMethod(char cArray[]);`
- E. All of the above

Passing Arrays



cs6394q, Peer 6- Java Arrays - Fall Semester 2016

Question -5





Which invocation correctly passes the character array *cArray* to the method?

- A. `myMethod(cArray[]);`
- B. `myMethod(cArray);`
- C. `myMethod(cArray[8]);`
- D. `myMethod(char cArray[]);`
- E. All of the above

Passing Arrays



cs6394q, Peer 6 - Java Arrays - Fall Semester 2016

Question - 6





Select the best description for the following code which 'copies' one array to another:

```
// Code fragment
char array0[] =
    {'a', 'b', 'c'};
char array1[] =
    new char[3];
array1 = array0;
```

- A. Copies the contents of array0 to array1
- B. Copies the reference to array0 to array1
- C. Copies the name of array0 to array1
- D. Does not even compile!

Array Copying



cs6394q, Peer 6 - Java Arrays - Fall Semester 2016

Question - 7





Select the best description for the following code which 'copies' one array to another:

```
// Code fragment
char array0[] =
    {'a', 'b', 'c'};
char array1[] =
    new char[3];
array1 = array0;
```

- A. Copies the contents of array0 to array1
- B. Copies the reference to array0 to array1
- C. Copies the name of array0 to array1
- D. Does not even compile!

Array Copying



cs6394q, Peer 6 - Java Arrays - Fall Semester 2016

Question - 8


