



Fill in the table below with a test suite that provides statement coverage. In the Covers column, list the letter labels (A, B, C, etc.) of the nodes covered by each test case.

Input		Expected Output	Covers
month	day		

The function is correct to the best of my knowledge.

What change to a line in the function would introduce a bug that your above test suite catches?

---



---



---



---

What change to a line in the function would introduce a bug that your above test suite does not catch?

---



---



---



---

Fill in the table below with a test suite that provides branch coverage. In the Covers column, list the number labels (1, 2, 3, etc.) of the edges covered by each test case (only true/false edges needed).

Input		Expected Output	Covers
month	day		

The function is correct to the best of my knowledge.

What change to a line in the function would introduce a bug that your above test suite catches?

---



---



---



---

What change to a line in the function would introduce a bug that your above test suite does not catch?

---



---



---



---

Fill in the table below with a test suite that provides path coverage. In the Covers column, list the number labels (1, 2, 3, etc.) of the edges covered by each test case. You need only cover executions that involve at most 1 iteration of each loop (if there are any).

Input		Expected Output	Covers
month	day		

The function is correct to the best of my knowledge.

What change to a line in the function would introduce a bug that your above test suite catches?

---



---



---



---

What change to a line in the function would introduce a bug that your above test suite does not catch?

---



---



---



---



Fill in the table below with a test suite that provides statement coverage. In the Covers column, list the letter labels (A, B, C, etc.) of the nodes covered by each test case.

Input			Expected Output	Covers
x	y	z		

The function is correct to the best of my knowledge.

What change to a line in the function would introduce a bug that your above test suite catches?

---



---



---



---

What change to a line in the function would introduce a bug that your above test suite does not catch?

---



---



---



---

Fill in the table below with a test suite that provides branch coverage. In the Covers column, list the number labels (1, 2, 3, etc.) of the edges covered by each test case (only true/false edges needed).

Input			Expected Output	Covers
x	y	z		

The function is correct to the best of my knowledge.

What change to a line in the function would introduce a bug that your above test suite catches?

---



---



---



---

What change to a line in the function would introduce a bug that your above test suite does not catch?

---



---



---



---

Fill in the table below with a test suite that provides path coverage. In the Covers column, list the number labels (1, 2, 3, etc.) of the edges covered by each test case. You need only cover executions that involve at most 1 iteration of each loop (if there are any).

Input			Expected Output	Covers
x	y	z		

The function is correct to the best of my knowledge.

What change to a line in the function would introduce a bug that your above test suite catches?

---



---



---



---

What change to a line in the function would introduce a bug that your above test suite does not catch?

---



---



---



---



Fill in the table below with a test suite that provides statement coverage. In the Covers column, list the letter labels (A, B, C, etc.) of the nodes covered by each test case.

Input		Expected Output	Covers
x	y		

The function is correct to the best of my knowledge.

What change to a line in the function would introduce a bug that your above test suite catches?

---



---



---



---

What change to a line in the function would introduce a bug that your above test suite does not catch?

---



---



---



---

Fill in the table below with a test suite that provides branch coverage. In the Covers column, list the number labels (1, 2, 3, etc.) of the edges covered by each test case (only true/false edges needed).

Input		Expected Output	Covers
x	y		

The function is correct to the best of my knowledge.

What change to a line in the function would introduce a bug that your above test suite catches?

---



---



---



---

What change to a line in the function would introduce a bug that your above test suite does not catch?

---



---



---



---

Fill in the table below with a test suite that provides path coverage. In the Covers column, list the number labels (1, 2, 3, etc.) of the edges covered by each test case. You need only cover executions that involve at most 1 iteration of each loop (if there are any).

Input		Expected Output	Covers
x	y		

The function is correct to the best of my knowledge.

What change to a line in the function would introduce a bug that your above test suite catches?

---



---



---



---

What change to a line in the function would introduce a bug that your above test suite does not catch?

---



---



---



---



Fill in the table below with a test suite that provides statement coverage. In the Covers column, list the letter labels (A, B, C, etc.) of the nodes covered by each test case.

Input		Expected Output	Covers
array	key		

The function is correct to the best of my knowledge.

What change to a line in the function would introduce a bug that your above test suite catches?

---



---



---



---

What change to a line in the function would introduce a bug that your above test suite does not catch?

---



---



---



---

Fill in the table below with a test suite that provides branch coverage. In the Covers column, list the number labels (1, 2, 3, etc.) of the edges covered by each test case (only true/false edges needed).

Input		Expected Output	Covers
array	key		

The function is correct to the best of my knowledge.

What change to a line in the function would introduce a bug that your above test suite catches?

---



---



---



---

What change to a line in the function would introduce a bug that your above test suite does not catch?

---



---



---



---

Fill in the table below with a test suite that provides path coverage. In the Covers column, list the number labels (1, 2, 3, etc.) of the edges covered by each test case. You need only cover executions that involve at most 1 iteration of each loop (if there are any).

Input		Expected Output	Covers
array	key		

The function is correct to the best of my knowledge.

What change to a line in the function would introduce a bug that your above test suite catches?

---



---



---



---

What change to a line in the function would introduce a bug that your above test suite does not catch?

---



---



---



---