

Zdá sa, že váš testovací prípad **test\_haversine** zlyhal. Tvrdenie naznačuje, že vypočítaná hodnota haversínu nezodpovedá očakávanej hodnote. Očakávaná hodnota je **945793.4375088713**, ale skutočná hodnota je **946.3876221719836**.

Tu je niekoľko krokov, pomocou ktorých môžete problém vyriešiť a vyriešiť:

- **Skontrolujte implementáciu Haversinovho vzorca** : Skontrolujte implementáciu svojej **haversine** funkcie v **myfunctions.py**. Uistite sa, že vzorec je správny a že pre vstupné súradnice používate správne jednotky (napr. radiány alebo stupne).
- **Skontrolujte argumenty funkcie** : Uistite sa, že argumenty odovzdané **haversine** funkcii vo vašom testovacom prípade sú správne. Zdá sa, že testovací prípad používa súradnice pre Amsterdam (52.370216, 4.895168) a Berlín (52.520008, 13.404954). Potvrďte, že tieto súradnice sú presné.
- **Aktualizujte testovacie očakávanie** : Ak ste potvrdili, že implementácia Haversinovho vzorca je správna, možno budete musieť aktualizovať očakávanú hodnotu vo vašom testovacom prípade, aby zodpovedala skutočnému výsledku. Je možné, že očakávaná hodnota v teste je zastaraná alebo nesprávna.

Tu je príklad, ako môžete aktualizovať test:

```
# Update the expected value in tests/test_myfunctions.py
assert myfunctions.haversine(52.370216, 4.895168, 52.520008, 13.404954)
== 946.3876221719836
```

- **Zvážte toleranciu s pohyblivou rádovou čiarkou** : Vzhľadom na povahu aritmetiky s pohyblivou rádovou čiarkou je bežné používať toleranciu pri porovnávaní čísel s pohyblivou rádovou čiarkou. Možno budete chcieť použiť tvrdenie, ktoré umožňuje malý rozdiel, napríklad:

```
assert abs(myfunctions.haversine(52.370216, 4.895168, 52.520008,
13.404954) - 945793.4375088713) < 1e-6
```

Upravte toleranciu ( **1e-6** v tomto prípade) na základe presnosti, ktorú očakávate vo svojich výpočtoch. Prejdením týchto krokov by ste mali byť schopní identifikovať a vyriešiť problém s vaším neúspešným testovacím prípadom.

Zdá sa, že váš testovací prípad **test\_haversine** zlyhal. Chyba tvrdenia naznačuje, že vypočítaný výsledok ( **946.3876221719836**) sa nezhoduje s očakávaným výsledkom (**945793.4375088713**). To naznačuje, že môže nastať problém s funkciou **haversine** vo vašom **myfunctions** module.

Mali by ste skontrolovať implementáciu funkcie **haversine** a skontrolovať, či produkuje očakávaný výstup. Tu je príklad, ako **haversine** môže funkcia vyzeráť:

```
# myfunctions.py
```

```
import math
```

```
def haversine(lat1, lon1, lat2, lon2):
    # Radius of the Earth in kilometers
    R = 6371.0

    # Convert latitude and longitude from degrees to radians
    lat1, lon1, lat2, lon2 = map(math.radians, [lat1, lon1, lat2, lon2])

    # Calculate the differences
    dlat = lat2 - lat1
    dlon = lon2 - lon1

    # Haversine formula
    a = math.sin(dlat / 2)**2 + math.cos(lat1) * math.cos(lat2) * math.sin(dlon / 2)**2
    c = 2 * math.atan2(math.sqrt(a), math.sqrt(1 - a))

    # Calculate the distance
    distance = R * c

    return distance
```

Uistite sa, že vaša implementácia zodpovedá tejto logike. Ak nájdete nejaké problémy, opravte ich a spustite testy znova.

Okrem toho, ak sú testovacie údaje presné a **haversine** funkcia je správna, je možné, že sa vyskytol problém so samotným testovacím prípadom. Dvakrát skontrolujte očakávaný výsledok a hodnoty, ktoré odovzdávate funkcii.

Ak problém pretrváva, uveďte kód funkcie **haversine**a testovací prípad ( **test\_haversine**), aby sme vám mohli ďalej pomôcť pri identifikácii a riešení problému.