Homework # 6

01286120 Elementary Systems Programming

Software Engineering Program

Faculty of Engineering, KMITL

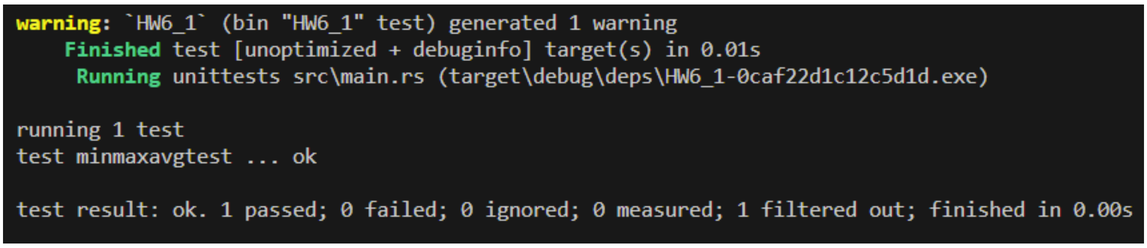
By

66011149 Phatthadon Sornplang

1. Write a function that calculate values from a list of numbers

1.1) Write the function min_max_avg to calculate the minimum, maximum, and average of the numbers within a list. Use loop to calculate values and return a tuple from the function. Create the unit test for the function to cover use cases.

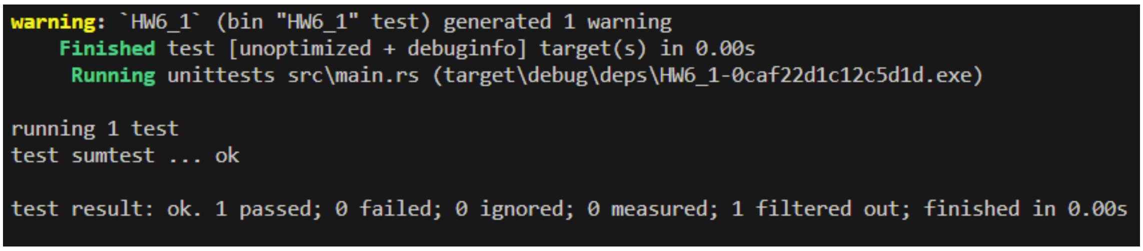
(Code in the attached file)



```
warning: `HW6_1` (bin "HW6_1" test) generated 1 warning
    Finished test [unoptimized + debuginfo] target(s) in 0.01s
     Running unittests src\main.rs (target\debug\deps\HW6_1-0caf22d1c12c5d1d.exe)

running 1 test
test minmaxavgtest ... ok

test result: ok. 1 passed; 0 failed; 0 ignored; 0 measured; 1 filtered out; finished in 0.00s
```

1.2) Write the function cal_partial_sums to calculate the partial sums of a number list. For example, cal_partial_sums(&[2, 11, 3, 4, 7]) would gives [2, 13, 16, 20, 27] . Use loop to calculate values and return a vector from the function. Create the unit test for the function to cover use cases.

(Code in the attached file)

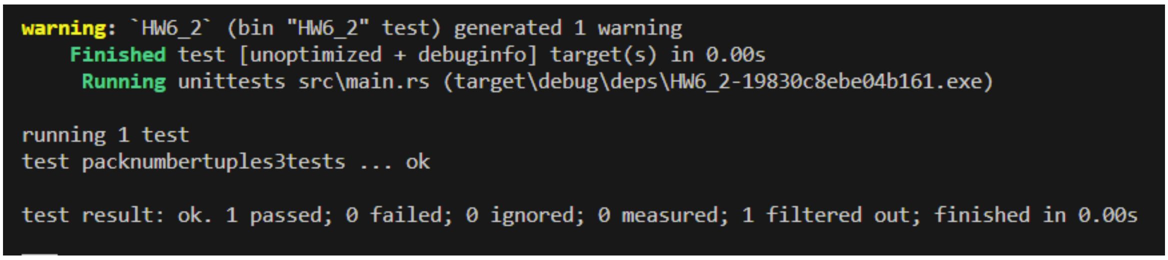


```
warning: `HW6_1` (bin "HW6_1" test) generated 1 warning
    Finished test [unoptimized + debuginfo] target(s) in 0.00s
     Running unittests src\main.rs (target\debug\deps\HW6_1-0caf22d1c12c5d1d.exe)

running 1 test
test sumtest ... ok

test result: ok. 1 passed; 0 failed; 0 ignored; 0 measured; 1 filtered out; finished in 0.00s
```

2. Write a function that combine two or more lists of numbers into a list of tuples

2.1) Write the function pack_number_tuples3 to combine three lists of numbers into a list of tuples (instead of two lists like in the lab exercise 4.1)). Add zero to fill up missing values from the shorter input number list to make the number of tuples equal to the longest number list. For example, pack_number_tuples3(&[1, 2], &[4, 3], &[5]) would gives [(1, 4, 5), (2, 3, 0)] . Create the unit test for the function to cover use cases.

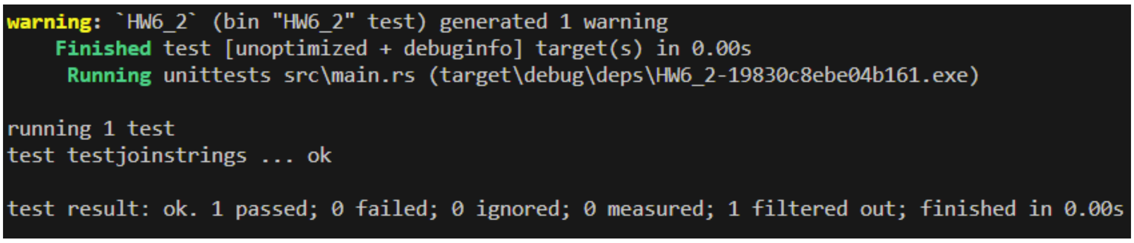
(Code in the attached file)



```
warning: `HW6_2` (bin "HW6_2" test) generated 1 warning
    Finished test [unoptimized + debuginfo] target(s) in 0.00s
     Running unittests src\main.rs (target\debug\deps\HW6_2-19830c8ebe04b161.exe)

running 1 test
test packnumbertuples3tests ... ok

test result: ok. 1 passed; 0 failed; 0 ignored; 0 measured; 1 filtered out; finished in 0.00s
```

2.2) Write the function pack_number_tuples_s3 to combine three lists of numbers into a list of tuples (instead of two lists like exercise 4.2) of this lab). The number of tuples will be equal to the shortest number list. For example, pack_number_tuples_s3(&[1, 2], &[4, 3], &[5]) would gives [(1, 4, 5)] . Create the unit test for the function to cover use cases. #[test] fn test_join_strings() { assert_eq!(pack_number_tuples_s(&[], &[]), []); assert_eq!(pack_number_tuples_s(&[1], &[]), []); assert_eq!(pack_number_tuples_s(&[], &[2, 3]), []); assert_eq!(pack_number_tuples_s(&[5, 1, 4], &[2, 3]), [(5, 2), (1, 3)]); } 01286120 Elementary Systems Programming 11

(Code in the attached file)

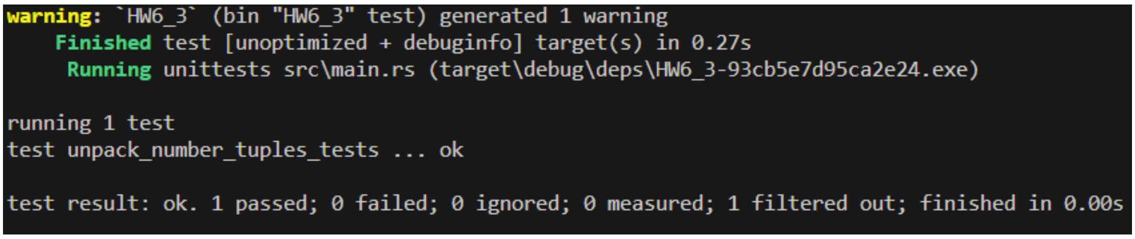


```
warning: `HW6_2` (bin "HW6_2" test) generated 1 warning
    Finished test [unoptimized + debuginfo] target(s) in 0.00s
     Running unittests src\main.rs (target\debug\deps\HW6_2-19830c8ebe04b161.exe)

running 1 test
test testjoinstrings ... ok

test result: ok. 1 passed; 0 failed; 0 ignored; 0 measured; 1 filtered out; finished in 0.00s
```

3. Write a function that extract two or more lists of numbers from a list of tuples

3.1) Write the function unpack_number_tuples to extract numbers from a list of tuples into two number lists. This operation would be the inverse of the function pack_number_tuples from exercise 4.1) of this lab. For example, unpack_number_tuples(&[(1, 4), (3, 2), (2, 1)]) would gives ([1, 3, 2], [4, 2, 1]) . Create the unit test for the function to cover use cases.

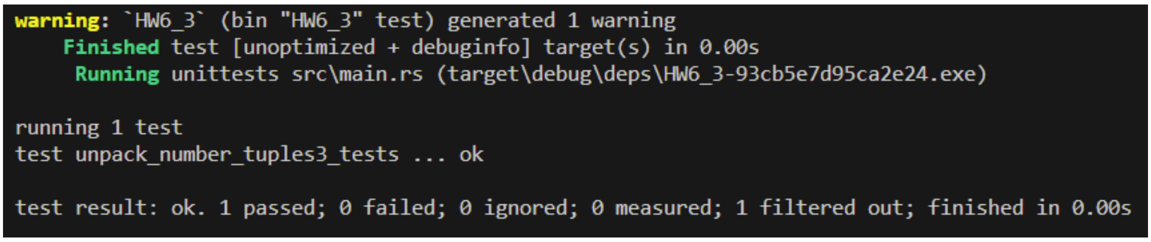
(Code in the attached file)



```
warning: `HW6_3` (bin "HW6_3" test) generated 1 warning
    Finished test [unoptimized + debuginfo] target(s) in 0.27s
     Running unittests src\main.rs (target\debug\deps\HW6_3-93cb5e7d95ca2e24.exe)

running 1 test
test unpack_number_tuples_tests ... ok

test result: ok. 1 passed; 0 failed; 0 ignored; 0 measured; 1 filtered out; finished in 0.00s
```

3.2) Write the function unpack_number_tuples3 to extract numbers from a list of tuples into three number lists. This operation would be the inverse of the function pack_number_tuples3 . For example, unpack_number_tuples3(&[(1, 4, 5), (2, 2, 1)]) would gives ([1, 2], [4, 2], [5, 1]) . Create the unit test for the function to cover use cases.

(Code in the attached file)



```
warning: `HW6_3` (bin "HW6_3" test) generated 1 warning
    Finished test [unoptimized + debuginfo] target(s) in 0.00s
     Running unittests src\main.rs (target\debug\deps\HW6_3-93cb5e7d95ca2e24.exe)

running 1 test
test unpack_number_tuples3_tests ... ok

test result: ok. 1 passed; 0 failed; 0 ignored; 0 measured; 1 filtered out; finished in 0.00s
```