UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE AND ENGINEERING

ECE 1724-F1 — Performant Software Systems with Rust

Midterm Examination — Answer Key

Coverage: Rust Book Chapters 1–6, 8, and 9.

40 questions; 100 marks; 110 minutes. Questions 1–20 carry 2 marks each; Questions 21–40 carry 3 marks each.

Coverage follows The Rust Programming Language, Chapters 1–6, 8, and 9.

Answer summary

1	C	11	C	21	B	31	C
2	A	12	B	22	D	32	B
3	D	13	D	23	A	33	D
4	B	14	A	24	C	34	A
5	B	15	C	25	C	35	C
6	D	16	B	26	A	36	A
7	A	17	A	27	D	37	D
8	C	18	D	28	B	38	B
9	A	19	C	29	D	39	C
10	D	20	B	30	A	40	B

UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE AND ENGINEERING

ECE 1724-F1 — Performant Software Systems with Rust

Midterm Examination — Answer Key (continued)

Coverage: Rust Book Chapters 1–6, 8, and 9.

Question 1: Ⓒ **2 marks**

cargo new station creates a project directory containing the package manifest and initial source code. *Book: 1.3.*

Question 2: Ⓐ **2 marks**

Editing source code does not change an already compiled executable. Another build is needed to incorporate the edit. *Book: 1.2–1.3.*

Question 3: Ⓓ **2 marks**

read_line takes a mutable reference so it can append the bytes read to the existing String. Ownership remains with the caller. *Book: 2.*

Question 4: Ⓑ **2 marks**

The second let shadows the String binding with a u32 obtained by parsing. Numeric addition then produces 13. *Book: 2.*

Question 5: Ⓑ **2 marks**

The inner binding shadows the outer binding only within the block. The outer value remains 3. *Book: 3.1.*

Question 6: Ⓓ **2 marks**

The loop reaches count 3 and break supplies 6 as the value of the loop expression. *Book: 3.5.*

Question 7: Ⓐ **2 marks**

The range includes 1, 2, and 3 but excludes 4, so the sum is 6. *Book: 3.5.*

Question 8: Ⓒ **2 marks**

Ownership moves into send_back and is returned to the new message binding after the text is extended. *Book: 4.1.*

UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE AND ENGINEERING

ECE 1724-F1 — Performant Software Systems with Rust

Midterm Examination — Answer Key (continued)

Coverage: Rust Book Chapters 1–6, 8, and 9.

Question 9: Ⓐ **2 marks**

saved is used after the mutation, so its immutable borrow overlaps the mutable borrow needed by push_str. Rust rejects the overlap. *Book: 4.2.*

Question 10: Ⓓ **2 marks**

The slice contains the elements at indices 1 and 2. It has length 2, and its first element is 20. *Book: 4.3.*

Question 11: Ⓒ **2 marks**

The update copies the u32 column field. Both fields have Copy types, so start remains usable; next has row 4 and column 5. *Book: 5.1.*

Question 12: Ⓑ **2 marks**

reset borrows the Counter mutably and changes its value field to zero. The counter remains available after the method call. *Book: 5.3.*

Question 13: Ⓓ **2 marks**

An enum value represents one variant. Rust allows variants with no data, tuple-style data, or named fields. *Book: 6.1.*

Question 14: Ⓐ **2 marks**

The other arm matches any remaining value and binds that value. Adding 2 produces 9 without changing original. *Book: 6.2.*

Question 15: Ⓒ **2 marks**

Some(0) is present data, distinct from None. The pattern matches and binds amount to zero. *Book: 6.3.*

Question 16: Ⓑ **2 marks**

get returns None when the index is outside the vector. Unlike indexing with brackets, this access does not panic. *Book: 8.1.*

UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE AND ENGINEERING

ECE 1724-F1 — Performant Software Systems with Rust

Midterm Examination — Answer Key (continued)

Coverage: Rust Book Chapters 1–6, 8, and 9.

Question 17: Ⓐ **2 marks**

format! builds a new String without taking ownership of left or right. left remains `north`. *Book: 8.2.*

Question 18: Ⓓ **2 marks**

String::len counts UTF-8 bytes: two for é and one for !. chars produces two characters. *Book: 8.2.*

Question 19: Ⓒ **2 marks**

The tram entry is vacant, so or_insert inserts 6. The map then contains bus and tram, giving length 2. *Book: 8.3.*

Question 20: Ⓑ **2 marks**

The first println executes. unwrap on Err panics before value is initialized, so the final println is never reached. *Book: 9.2.*

Question 21: Ⓑ **3 marks**

Zero resets the current run. The positive runs have lengths 2, 3, and 1, so there are three groups, a longest run of three, and a final run of one. *Book: 3.5.*

Question 22: Ⓓ **3 marks**

Digits 3, 0, and 4 are appended to reversed. On the fourth iteration the remaining digit is 2; break returns it before it can be appended. *Book: 3.2; 3.5.*

Question 23: Ⓐ **3 marks**

The clone remains `box`. The first call records length 3 before appending `!`. Appending the clone makes `box!box` (length 7), and the second call adds one more `!`. The shared borrow ends before later mutation. *Book: 4.1, 4.2.*

Question 24: Ⓒ **3 marks**

Passing original transfers ownership to the function. The function modifies and returns that String, whose ownership then belongs to chosen. The final use of original is invalid, so compilation fails and no lines print. The clone backup is independently owned. *Book: 4.1.*

UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE AND ENGINEERING

ECE 1724-F1 — Performant Software Systems with Rust

Midterm Examination — Answer Key (continued)

Coverage: Rust Book Chapters 1–6, 8, and 9.

Question 25: Ⓒ **3 marks**

Each colon advances start, so suffix returns the text after the last colon. The first slice prints `def` and its saved length is 3. Its borrow is no longer used when path is extended; the new suffix is `xy`. *Book: 4.2, 4.3.*

Question 26: Ⓐ **3 marks**

The first transfer changes the balances to 5 and 6. Sending 6 from a balance of 5 returns false without changing either wallet. Sending 3 back produces 8 and 3. *Book: 5.1, 5.3.*

Question 27: Ⓓ **3 marks**

Struct update moves the String address but copies the u32 weight. The bool express is supplied explicitly in second. The remaining Copy fields of first can still be accessed; mutations of second do not change them. *Book: 4.1, 5.1.*

Question 28: Ⓑ **3 marks**

widened borrows original and constructs a separate rectangle of dimensions 5 by 4. Only expanded is changed to height 5, giving areas 12 and 25. *Book: 5.1, 5.3.*

Question 29: Ⓓ **3 marks**

The first save records 4. Adding 7 is undone by restoring that saved value. Adding 2 then saving produces current and saved values of 6, with two saves. *Book: 6.1, 6.2.*

Question 30: Ⓐ **3 marks**

The adjacent valid pairs are 3 to 8 and 5 to 7, contributing 5 and 2. None clears previous, so there is no pair between 8 and 5. *Book: 6.2, 6.3.*

Question 31: Ⓒ **3 marks**

The four calls return Some(7), None, Some(10), and Some(0). Zero is a valid price, so only one coupon is rejected and the total is 17. *Book: 6.1, 6.2.*

Question 32: Ⓑ **3 marks**

pop removes from the end. The values processed are 6, then the newly pushed 1, then 4, then 2. Their sum is 13 and the vector is empty after four removals. *Book: 6.2, 8.1.*

UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE AND ENGINEERING

ECE 1724-F1 — Performant Software Systems with Rust

Midterm Examination — Answer Key (continued)

Coverage: Rust Book Chapters 1–6, 8, and 9.

Question 33: Ⓓ **3 marks**

The first loop replaces the vector with prefix sums [3, 4, 8, 10]. Appending 10 gives three values at least 7: 8, 10, and 10. Their sum is 28; index 1 contains 4. *Book: 8.1.*

Question 34: Ⓐ **3 marks**

Price 6 is accepted under the initial limit 10, and Free increases only the count. The limit becomes 4, so price 7 is skipped and price 2 is accepted: total 8, count 3. *Book: 6.2, 8.1.*

Question 35: Ⓒ **3 marks**

Empty words are skipped before separators are added, leaving three words. format! creates a separate String without taking ownership of line, so appending ! changes only line. *Book: 3.5, 8.2.*

Question 36: Ⓐ **3 marks**

The pen orders of 3 and 2 are filled, reducing pen stock to zero. The pad order is rejected without reducing its stock. entry inserts a zero-valued clip key even though that order cannot be filled, so the map has three keys. *Book: 8.3.*

Question 37: Ⓓ **3 marks**

The first pass gives up=3, down=2, left=1. insert replaces up with 1. The second pass gives left=2, right=4 (starting at 2 only on insertion), and up=2. The order-independent sum is 10. *Book: 8.3.*

Question 38: Ⓑ **3 marks**

The first value prints `added 2`. Parsing bad makes ? return Err immediately, so neither the third value nor a subtotal is processed. main takes the error branch and changes saved from 10 to 9. *Book: 9.2.*

Question 39: Ⓒ **3 marks**

Valid nonnegative values 4, 0, and 3 sum to 7. Both the parsed negative value and the invalid numeric text become recoverable errors, so rejected is 2 and the loop continues. *Book: 9.2.*

Question 40: Ⓑ **3 marks**

The borrowed match inspects each Result without moving it. The first expect returns 3. On bad, the match prints `retry needed`, but expect then panics; the third input and final `done` line are never reached. *Book: 6.2, 9.2.*