LLM Transpilation Bug Analysis by Problem

Summary Table

Problem	Total Bugs	Category 1 (Register)	Category 2 (Control Flow)	Category 3 (Omission)	Category 4 (Instruction)	Category 5 (x86-isms)
6	1	0	0	1	0	0
7	1	0	0	0	1	0
8	1	1	0	0	0	0
9	3	1	0	2	0	0
10	1	0	1	0	0	0
15	2	0	0	0	1	1
18	4	1	1	0	2	0
19	1	1	0	0	0	0
20	3	1	0	0	1	1
TOTALS	**17**	**5**	**2**	**3**	**5**	**2**

Detailed Problem Breakdown

Problem 6 (1 bug)

Category	Pattern	Error Description
Category 3	Omission	SIMD loop setup incomplete - missing array length to iteration count conversion

Problem 7 (1 bug)

Category	Pattern	Error Description
Category 4	4.3 - Incorrect Ordering	Resets `max_level = 0` before storing value, always stores 0

Problem 8 (1 bug)

Category	Pattern	Error Description
Category 1	1.1 - Premature Update	Increments counter before use, writes to `array[count+1]` instead of `array[count]`

Problem 9 (3 bugs)

Category	Pattern	Error Description
Category 3	Omission	Missing initialization of three SIMD vector registers
Category 3	Omission	Missing final SIMD reduction step for product calculation
Category 1	1.2 - Register Clobbering	Reuses sum vectors as multiplication destinations, destroys sum data

Problem 10 (1 bug)

Category	Pattern	Error Description
Category 2	2.2 - Incorrect Return	Premature exit, returns integer cast as pointer instead of array pointer

Problem 15 (2 bugs)

Category	Pattern	Error Description
Category 5	x86-ism	Uses complex x86 bit-shifting for `N * 8` instead of simple ARMv8 `lsl`/`sbfiz`
Category 4	4.1 - Invalid Addressing	Uses same register `x0` as both base and offset: `[x0, x0]`

Problem 18 (4 bugs)

Category	Pattern	Error Description
Category 4	4.2 - Incorrect Constants	Wrong stack offsets: `[sp, #14]` instead of `[sp, #12]`
Category 4	4.2 - Incorrect Constants	Stores wrong value: `4` instead of `1` for musical note type
Category 1	1.3 - Return Value Failure	Missing `mov x21, x0` after `realloc`, leaves stale pointer
Category 2	2.1 - Incorrect Branch	Branches to loop block instead of error handler after `realloc` failure

Problem 19 (1 bug)

Category	Pattern	Error Description
Category 1	1.2 - Register Clobbering	Overwrites loop counter with input value, loop executes only once

Problem 20 (3 bugs)

Category	Pattern	Error Description
Category 1	1.2 - Register Clobbering	Overwrites original character with lowercase version, corrupts output
Category 4	4.2 - Incorrect Constants	Wrong ASCII values in character classification comparisons
Category 5	x86-ism	Unnecessary stack allocation and x86-style register spill/fill operations

Category Distribution Analysis

Category	Count	Percentage	Most Common Pattern
Category 1 - Register State	5	29.4%	Register Clobbering (3/5)
Category 4 - Instruction Semantics	5	29.4%	Incorrect Constants (3/5)
Category 3 - Critical Omissions	3	17.6%	Missing operations
Category 2 - Control Flow	2	11.8%	Split between patterns
Category 5 - x86-isms	2	11.8%	Non-idiomatic translations

Key Insights

- **Most problematic areas**: Register management and instruction semantics (58.8% combined)
- **Highest bug density**: Problem 18 (4 bugs) and Problem 9 (3 bugs)
- **Most frequent specific error**: Register clobbering (3 instances across 3 problems)
- **Critical pattern**: Problems with memory allocation functions (realloc/malloc) consistently show multiple error types