

Prompt

Task: Orchestrate ARVO bug localization and evaluation.

Follow this high-level pipeline:  
(1) Choose REPO\_ROOT:  
...

Thought

Planner note

Observation

You are the PLANNER for ARVO bug localization. You orchestrate the pipeline, call ...

Thought

Planner note

Observation

Task: Orchestrate ARVO bug localization and evaluation.

Follow this high-level pipeline:  
(1) Choose REPO\_ROOT:  
...

Thought

Planner call: chunk\_case

```
$ chunk_case(root_dir="/src/libreoffice",  
out_json_path="/home/hx759  
bug_localization_agents/work_arvo/16614  
case_chunks.json")
```

Thought

Planner result: chunk\_case

chunk the repo into 5441 chunks for  
evaluation

Observation

```
{"ok": true, "root_dir": "/src/libreoffice", "json_path":  
"/home/hx759/bug_localization_agents/work_arvo/16614  
case_chunks.json", "num_files": 469, "num_chunks": 5441}
```

Thought

Planner call: diff\_index

```
$  
diff_index(diff_path="/home/hx759  
bug_localization_agents/ARVO-Meta/patches  
16614.diff",  
out_json_path="/home/hx759  
bug_localization_agents/work_arvo/16614  
diff_index.json")
```

Thought

Planner result: diff\_index

Observation

```
{"ok": true, "json_path":  
"/home/hx759/bug_localization_agents/work_arvo/16614  
diff_index.json", "num_files": 1, "num_hunks": 0,  
"num_per_line": 4}
```

Thought

Planner call: mark\_diff

```
$  
mark_diff(diff_json_path="/home/hx759  
bug_localization_agents/work_arvo/16614  
diff_index.json",  
chunks_json_path="/home/hx759  
bug_localization_agents/work_arvo/16614  
case_chunks.json")
```



Thought

Planner result: mark\_diff

### Observation

```
{"ok": true, "chunks_json_path":  
"/home/hx759/bug_localization_agents/work_arvo/16614  
case_chunks.json", "num_chunks": 5441,  
"num_chunks_with_diff": 1}
```

Thought

Planner call: run\_san

```
$ run_san()
```

Thought

Planner result: run\_san

### Observation

```
{"ok": true, "asan_output": {"ok": true, "asan_output":  
"INFO: Seed: 1963705179\r\nINFO: Loaded 1 modules  
(1142 inline 8-bit counters): ...", "binary_path":  
"/out/fuzz_asn1_print", "cid": "9b78c01c42f0"}}
```

Thought

Planner call: delegate

```
$ delegate(task="llm_analyze() ->  
extract_json\nOptionally view source/grep source  
around high-confidence lines\nIf available,  
publish verified locations(repo root='/src'  
libreoffice')\nCompare llm metrics with  
diff_index_json_path='./diff_index.json',  
llm_json_path=<raw JSON>,  
chunks_json_path='./case_chunks.json',  
line_stack=0\nFinish task with summary including  
metrics and paths")
```

Thought

Planner result: delegate

### Observation

```
"ok=True; detection_rate=1.000; localization_rate=1.000;  
strict_localization_rate=0.000; improved=True;  
prev=1.000; curr=1.000;  
diff_index_json_path=./diff_index.json;  
chunks_json_path=./case_chunks.json;  
llm_json_path=/home/hx759/bug_localization_agents  
work_arvo/16614/llm_batch_0001.json; gdb=unknown;  
bin=unknown"
```

Thought

Executor#0 note

### Observation

You are the EXECUTOR for ARVO bug localization. You run heavy tools and MUST call finish\_task(summary=...) to return results.  
TOOLS:  
- llm\_analyze(crash\_log=<string> ...

Thought

Executor#0 note

### Observation

You have been delegated a task:  
Use the cached crash\_log produced by the Planner.  
Analyze, optionally augment with source views, ...

Thought

Executor#0 call: llm\_analyze

```
$ llm_analyze()
```



Thought

Executor#0 result: llm\_analyze

### Observation

```
{"ok": true, "mode": "crash", "text": "[{\"file_path\": \"/src/opensc/src/libopensc/asn1.c\", \"start_line\": 573, \"end_line\": 598, \"confidence\": \"very likely\", \"reason\": \"Heap-buffer-overflow reported at decode bit string. Needs bounds checking on input size.\", \"symbol\": \"decode bit string\"}]", "logged": {"jsonl_path": "/home/hx759/bug_localization_agents/work_arvo/16614_predictions.jsonl", "batch_json_path": "/home/hx759/bug_localization_agents/work_arvo/16614_llm_batch_0001.json", "cumulative_path": "/home/hx759/bug_localization_agents/work_arvo/16614_cumulative_predictions.json", "count": 1, "batch": 1}}
```

Thought

Executor#0 call: extract\_json

```
$ extract_json(response_text="[{\"file_path\": \"/src/opensc/src/libopensc/asn1.c\", \"start_line\": 573, \"end_line\": 598, \"confidence\": \"very likely\", \"reason\": \"Heap-buffer-overflow reported at decode bit string. Needs bounds checking on input size.\", \"symbol\": \"decode bit string\"}]", merge_with_last=false)
```

Thought

Executor#0 result: extract\_json

### Observation

```
{"ok": true, "json": {"llm_predictions": [{"file_path": "/src/opensc/src/libopensc/asn1.c", "start_line": 573, "end_line": 598, "confidence": "very likely", "reason": "Heap-buffer-overflow reported at decode bit string. Needs bounds checking on input size.", "symbol": "decode bit string"}]}, "raw": "{\"llm_predictions\": [{\"file_path\": \"/src/opensc/src/libopensc/asn1.c\", \"start_line\": 573, \"end_line\": 598, \"confidence\": \"very likely\", \"reason\": \"Heap-buffer-overflow reported at decode bit string. Needs bounds checking on input size.\", \"symbol\": \"decode bit string\"}]}\", \"json_path\": \"/home/hx759/bug_localization_agents/work_arvo/16614_llm_output.tmp.json\"}
```

Thought

Executor#0 call: compare\_llm\_metrics

```
$ compare_llm_metrics(llm_json_path="/home/hx759/bug_localization_agents/work_arvo/16614_llm_batch_0001.json", chunks_json_path="/home/hx759/bug_localization_agents/work_arvo/16614_case_chunks.json", diff_index_json_path="/home/hx759/bug_localization_agents/work_arvo/16614_diff_index.json", line_slack=0)
```

Thought

Executor#0 result: compare\_llm\_metrics

### Observation

```
{"ok": true, "detection_rate": 1.0, "localization_rate": 1.0, "strict_localization_rate": 0.0, "inputs": {"llm_json_path": "/home/hx759/bug_localization_agents/work_arvo/16614_llm_batch_0001.json", "chunks_json_path": "/home/hx759/bug_localization_agents/work_arvo/16614_case_chunks.json", "diff_index_json_path": "/home/hx759/bug_localization_agents/work_arvo/16614_diff_index.json"}, "detection": {"denom_diff_chunks": 1, "num_detected_and_diff": 1, "rate": 1.0, "updated_chunks": true}, "localization": {"total_lines": 4, "matched_lines": 4, "newly_matched": 4, "rate": 1.0}, "strict_localization": {"total_groups": 2, "matched_groups": 0, "rate": 0.0}, "saved_snapshot_error": "compare_history"}
```