

SA Homework 1

Shell Script

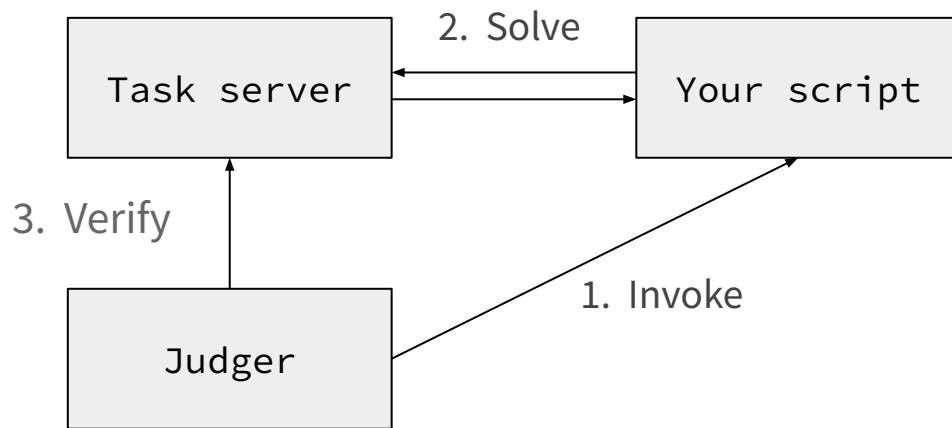
ymlai

Requirements

- In homework 1, you need to write a shell script to complete the tasks from the task server.
- Judger will log in to your server via ssh and invoke the script. If you send the wrong answer to the task server (or timeout), you will not get the points of the checkpoint.
- Hint:
 - CLI tools: curl, jq, awk, grep...

Task Server

- Create tasks and validate the problem answer
- Endpoint: <http://192.168.255.69>
- Flow:



Task Server API (1/6)

- GET /
 - Response
 - Body (application/json)

```
{  
  "message": "Have fun to 2025 NYCU SA HW1!"  
}
```

- GET /dictionary
 - install the dictionary.txt

```
aahed  
aalii  
aargh  
aarti  
.  
.  
.
```

Task Server API (2/6)

- POST /tasks
 - Request
 - Body (application/json)

```
{  
  "stuid": "75",  
  "type": "WORDLE"  
}
```

```
{  
  "stuid": "65",  
  "type": "QUORDLE"  
}
```

- Response
 - Body (application/json)

```
{  
  "stuid": "75",  
  "type": "WORDLE",  
  "id": "4a9c99f4-0241-4f3d-a003-7b2f6bb455db",  
  "status": "PENDING",  
  "guess": "0",  
  "problem": "XXXXX"  
}
```

```
{  
  "stuid": "65",  
  "type": "QUORDLE",  
  "id": "4a9c99f4-0241-4f3d-a003-7b2f6bb455db",  
  "status": "PENDING",  
  "guess": "0",  
  "problem1": "XXXXX",  
  "problem2": "XXXXX",  
  "problem3": "XXXXX",  
  "problem4": "XXXXX"  
}
```

Task Server API (3/6)

- POST /tasks/:id/submit
 - Request
 - Body (application/json)

```
{  
  "answer": "WATER"  
}
```

```
{  
  "answer": "MINOR"  
}
```

- Response
 - Body (application/json)

```
{  
  "problem": "XXBXA"  
}
```

```
{  
  "problem1": "XXBXA",  
  "problem2": "XXXXX",  
  "problem3": "XXBXX",  
  "problem4": "AXAXX"  
}
```

Task Server API (4/6)

- GET /tasks/:id
 - Response
 - Body (application/json)

```
{
  "stuid": "75",
  "type": "WORDLE",
  "id": "4a9c99f4-0241-4f3d-a003-7b2f6bb455db",
  "status": "PENDING",
  "guess": "0",
  "problem": "XXXXXX"
}
```

```
{
  "stuid": "65",
  "type": "QUORDLE",
  "id": "4a9c99f4-0241-4f3d-a003-7b2f6bb455db",
  "status": "PENDING",
  "guess": "0",
  "problem1": "XXXXXX",
  "problem2": "XXXXXX",
  "problem3": "XXXXXX",
  "problem4": "XXXXXX"
}
```

Task Server API (5/6)

- DELETE /tasks/:id
 - Response
 - Body (application/json)

```
{
  "stuid": "75",
  "type": "WORDLE",
  "id": "4a9c99f4-0241-4f3d-a003-7b2f6bb455db",
  "status": "FAILED",
  "guess": "10",
  "problem": "XXXBX"
}
```

```
{
  "stuid": "65",
  "type": "QUORDLE",
  "id": "4a9c99f4-0241-4f3d-a003-7b2f6bb455db",
  "status": "FAILED",
  "guess": "20",
  "problem1": "ABXB",
  "problem2": "XAXB",
  "problem3": "BBXA",
  "problem4": "XXBX"
}
```


Task Server API (6/6)

- GET /tasks/stu/:stuid
 - Response
 - Body (application/json)

```
{
  "stuid": "75",
  "tasks": [
    {
      "id": "4a9c99f4-0241-4f3d-a003-7b2f6bb455db",
      "type": "WORDLE"
    },
    {
      "id": "9c8d23a4-1b7e-4b9a-a41c-d2a91f73b8f4",
      "type": "WORDLE"
    },
    {
      "id": "9c8d23a4-1b7e-4b9a-a41c-d2a91f73b8f4",
      "type": "QUORDLE"
    }
  ]
}
```

Script Spec (1/2)

- Provide an executable shell script placed at `/home/judge/hw1.sh` with following available options:
 - `-t`
 - Task type
 - `"SYS_INFO"` | `"WORDLE"` | `"QUORDLE"`
 - `-h`
 - Optional
 - If the option is specified, just print the usage to `stderr`

Script Spec (2/2)

- Invalid arguments should be rejected with a non-zero status code, with the exact help message outputted to stderr

```
$ ./hw1.sh -a
hw1.sh -t TASK_TYPE [-h]

Available Options:
-t SYS_INFO | WORDLE | QUORDLE : Task type
-h : Show the script usage
```

Problem Spec - SYS_INFO

- Print the system information
 - Any extra tools or hardcode is forbidden

```
$ ./hw1.sh -t SYS_INFO
OS: Debian GNU/Linux 13 (trixie) x86_64
Kernel: 6.12.41+deb13-amd64
Shell: sh
Terminal: /dev/pts/0
CPU: Intel(R) Core(TM) i9-9880H CPU @ 2.30GHz
```

Problem Spec - WORDLE

1. Request a task from the server.
2. Solve the task in a single execution (the judge will ignore the output).

- Problem status

- A: The correct letter is on correct position
- B: The letter is in real answer but wrong position
- X: The letter is not in the answer

Example:

```

> ~/Desktop/sa ./hw1.sh -t WORDLE
{
  "stuid": "51",
  "type": "WORDLE",
  "id": "ef4bf6dc-4ba2-4cbf-a006-3d5866369e93",
  "status": "PENDING",
  "guess": "0",
  "problem": "XXXXX"
}
[+] Find Answer 'stedd'
{
  "stuid": "51",
  "type": "WORDLE",
  "id": "ef4bf6dc-4ba2-4cbf-a006-3d5866369e93",
  "status": "SOLVED",
  "guess": "5",
  "problem": "AAAAA"
}
```

reference: <https://www.nytimes.com/games/wordle/index.html>

Problem Spec - QUORDLE

1. Request a task from the server
2. Solve the task in a single execution (the judge will ignore the output)

Example:

- Problem status
 - A: The correct letter is on correct position
 - B: The letter is in real answer but wrong position
 - X: The letter is not in the answer

```

> ~/Desktop/sa ./hw1.sh -t QUORDLE
{
  "stuid": "51",
  "type": "QUORDLE",
  "id": "66da1a92-fc46-41f7-8568-2e87af6f13ff",
  "status": "PENDING",
  "guess": "0",
  "problem1": "XXXXX",
  "problem2": "XXXXX",
  "problem3": "XXXXX",
  "problem4": "XXXXX"
}

[*] Final Answers : trest hizen flock nites
{
  "stuid": "51",
  "type": "QUORDLE",
  "id": "66da1a92-fc46-41f7-8568-2e87af6f13ff",
  "status": "SOLVED",
  "guess": "9",
  "problem1": "AAAAA",
  "problem2": "AAAAA",
  "problem3": "AAAAA",
  "problem4": "AAAAA"
}

```

reference: <https://www.merriam-webster.com/games/quordle/#/>

Problem Status (Wordle / Quordle)

- SOLVED
 - The correct answer was found before the maximum guess limit was reached.
- FAILED
 - The maximum guess limit was reached without finding the correct answer.
- TIMEOUT
 - The problem remained unsolved after 10 seconds.
- PENDING
 - The only status in which guesses can be submitted.

Note for Wordle / Quordle

- stuid is calculated from your ip (ex: ip = 192.168.0.50, stuid = 0 * 256 + 50)
- The max guess limit of Wordle: 10
- The max guess limit of Quordle: 20
- The max tasks (wordle + quordle) of each user: 10
- Any output produced while solving Wordle or Quordle is acceptable
- There is a 30% chance of randomly generating **extra TIMEOUT tasks** (between 1 and 10) when a user sends a POST request to server/tasks
- Deleting or searching for other students' problems is forbidden. Any illegal request will be logged by the judge server.

Shellcheck

- ShellCheck is a static analysis tool for shell scripts. It helps catch errors and suggests improvements to make your scripts better.
 - `shellcheck -s sh -e SC2086 -a [your_script]` (should return zero)
 - Version : 0.10.0
 - Make sure you installed the shellcheck or the judge may failed
-
- Note: **Disable any checks in the file is forbidden**

SH(1): getopt

- `getopts optstring var`

The POSIX `getopts` command. The `getopts` command deprecates the older `getopt(1)` command. The first argument should be a series of letters, each possibly followed by a colon which indicates that the option takes an argument. The specified variable is set to the parsed option. The index of the next argument is placed into the shell variable `OPTIND`. If an option takes an argument, it is placed into the shell variable `OPTARG`.

Restrictions

- Must not use any other interpreters, compilers or programming languages (such as Python, Ruby, Node.js, Go, Rust, Perl, GCC, Clang...)
- Must not call any other self-written scripts, binaries or executables.
- Extra file is not permitted during execute the script (except dictionary).
- Only one shell, sh, is allowed.
- Common tools (e.g. date, openssl, jq, etc.) are allowed.
- If you are not sure whether a tool is allowed, please ask TA on Google Groups.
- Any violation will result in **0 points** for the assignment.

Grading

Automated grading (Online Judge), 100 pts

- Usage (15%)
 - Invalid options
 - Exit Code (5%)
 - Help Message (5%)
 - Invalid type (5%)
- Arbitrary argument position (10%)
- Shellcheck (10%)
- Tasks (65%)
 - sysinfo (10%)
 - wordle (25%)
 - quordle (30%)

Attention

- Your work will be scored by Online Judge system
 - Only the **LAST** submission will be scored
 - Late submission will **NOT** be accepted
- **ALWAYS BACKUP** your system before submission, as we may do malicious actions
- You are restricted to use only sh to complete your work
 - If you're not sure what's allowed, contact TAs.
 - TAs reserve the right of final explanations. Specs and the points of each subjudges are subject to change in any time.
- Make sure everything works after reboot

Rules

- TAs reserve the rights of final explanations
- **Open from 9/15 (Mon) 19:00**
- **Deadline: 10/13 (Mon) 23:59**
- Late submissions will **NOT** be accepted

Appendix - Useful Utilities

- [sh\(1p\)](#)
- [jq\(1\)](#)
- [curl\(1\)](#)
- [grep\(1\)](#)
- [awk\(1p\)](#)
- [sed\(1\)](#)
- [head\(1\)](#)
- [tail\(1\)](#)
- [seq\(1\)](#)
- [cut\(1\)](#)
- [sort\(1\)](#)
- [cat\(1\)](#)
- [echo\(1\)](#)
- [exit\(1p\)](#)