Homework 4

System Programming

May 17, 2021

Muhammed Okumuş

151044017

Contents

1 Functions

Under this section I will mention some of the important functions utilized in the program flow.

```c
// Parsing and printing.
void print_usage(void);
int lines(int fd);                        // Number of non-empty lines in the file
int chars(int fd);                        // Number of valid homeworks in the file
char *read_line(int fd, int n);           // Nth line in fle
void tsprintf(const char *format, ...); // Atomic printf

// Workers
void *student(void *data);            // Sleep, Code & Repeat
void *cheater(void *data);            // Tweet, Cheat & Repeat
void manager(void);                   // Main thread management

// Worker helpers
int have_enough_money(void);          // >0 if can atleast afford 1 student
int select_student(char hw_type);     // id of a available & most suitable student
int get_value(int i, char key);       // get C/S/Q of Nth student
void print_stats(void);               // End of program statics

// Wrappers
int s_wait(sem_t *sem);
int s_post(sem_t *sem);
int s_init(sem_t *sem, int val);

// Signal handler
void sig_handler(int sig_no);
```

Parsing are printing functions are fairly self-explanatory. Lines function is used to determine number of students in file since they are delimitered by newline. Chars function for determining the total number of homeworks in the file. Read line functions reads the Nth line in the file and returns it as character pointer. Thread safe printf(tsprintf) is a wrapper for printf function that utilizes a semaphore to make sure whole string is printed without getting effected by a context switch. Tsprintf is coded early in the design and probably not needed during the final version but left as is for future usage.

Worker functions are the functions used by the threads. There are multiple students but only one cheater and manager. Manager function is directly called in the main flow. Encapsulated main threads jobs to increase readability. These functions will be explained in detail in the next sections.

Worker helpers are also self-explanatory. They are used by the worker function to improve readablity and reduce code resuse. Wrappers are used to combine system calls with error checks and improve usability also.

2 Student Threads Data Struct

In this section data structed passed to the student threads are explained.

```c
struct Student
{
  char name[64];       // Name of the student
  int quality;         // Q
  int speed;           // S
  int cost;            // C

  //status variables
  int homeworks_done; // HW's completed by
  int money_made;     // Money made by
  int is_busy;        // Currently sleeping

  int pipe_fd[2];     // Pipe descriptors
};
```

The first four variables are populated from the students file. Next threee varibles are used to records the currents status of the student and homeworks done and money made is also used in the printing of the final statistics. 'Is busy' variable is set '1' while the studend sleep and checked by the select student function to check students availability.

3 Cheater Thread Function

This function is passed to the cheater tread and doesn't require a data pointer so void pointer is left a NULL. It carries homeworks from the homeworks file to a char array buffer for processing.

```c
void *cheater(void *data){
  int i = 0;
  char c;
  while (!exit_requested || !term_flag){
    wait(access);
    if (jobs_read >= max_jobs || !have_enough_money() || term_flag){
      // Print termination here ...
      post(jobs_read);
      post(access);
      return NULL;
    }
    // Valid homeworks & byte read
    pread(fd_homeworks, &c, 1, i++);
    if (c == 'Q' || c == 'S' || c == 'C'){
      // Print current stats here ...
      jobs[jobs_read++] = c;
      post(jobs_read);
    }
    post(access);
  }
  return NULL;
}
```

4 Manager Thread Function

This function acts as a middleman between the cheater and the student threads. Its job is to wait for a available homework in the queue put by the cheater and find a suitable student to solve the homework. After finding the suitable student, it communucates to the student via pipe. Syncronization between manager and the students are all mostly done via pipes. Students are also terminated by a pipe message send by the manager. This function is a bit long to put it in the report so I recommend checking the main.c file at lie 274.

5 Student Thread Function

This function is passed to the student threads and simulates solving a homework by sleeping. It takes orders from the manager function by a pipe declared in the Student struct passed as a void pointer. Also marks self available with a semaphore posted in this function and waited in the manager function.

6 Thread Creation

Only students threads are joined since the cheater thread is detached. Not detaching or joining a thread will cause a memory leak warning from valgrind.

```c
...
  for (int i = 0; i < n_students; i++)
    pthread_create(&thread_ids[i], NULL, student, &students[i]);

  pthread_create(&thread_ids[n_students], NULL, cheater, NULL);
  pthread_detach(thread_ids[n_students]);

  manager();
...
  // Join threads and free resources =================================
  for (int i = 0; i < n_students; i++)
    pthread_join(thread_ids[i], NULL);
```

7 Signal Handling

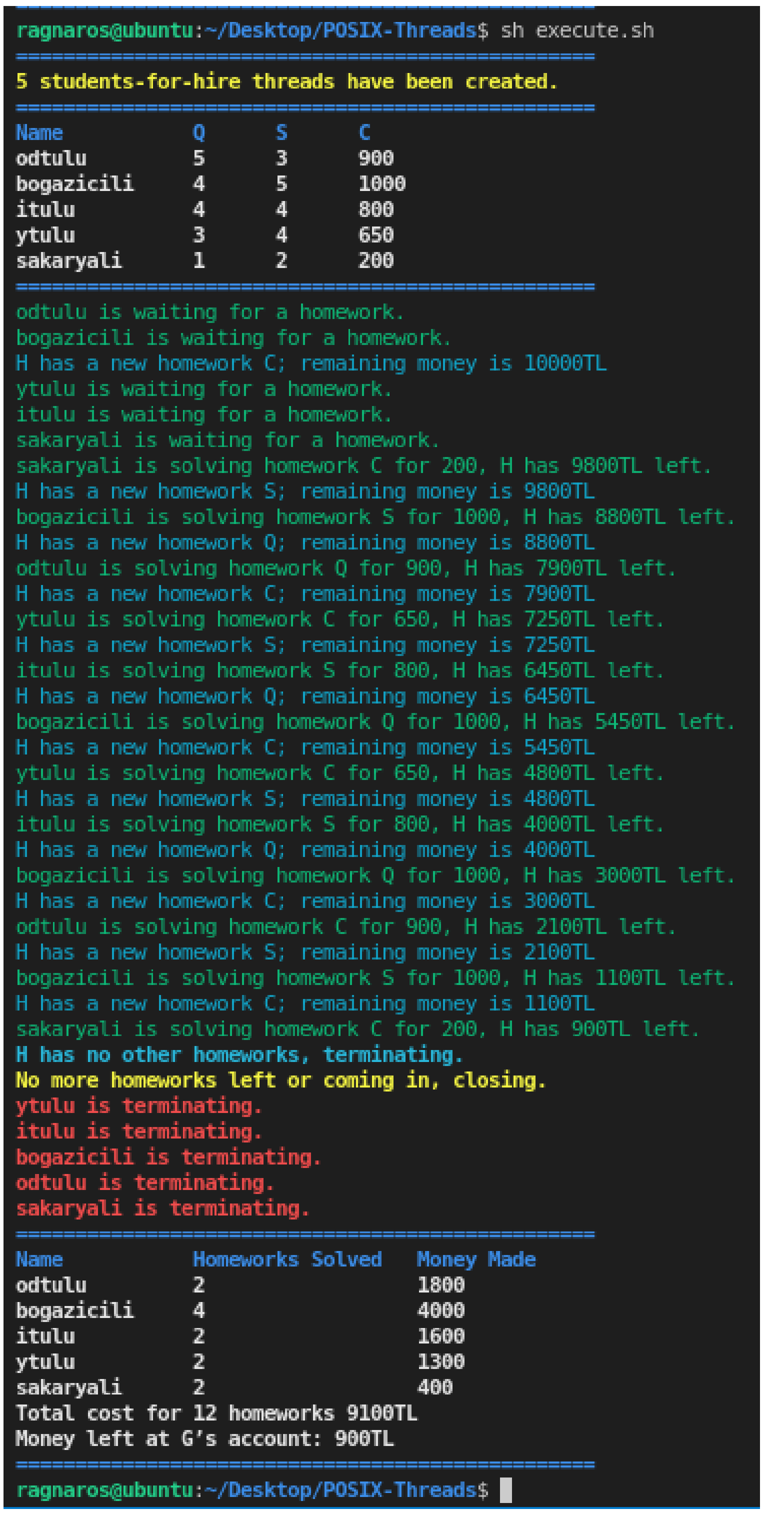
Signal handler is a simple setter function that sets an atomic variable that is check by the thread functions in their main 'while' loop. If it's set the functions fall trough for resource freeing code.

```c
  while(!exit_requested){
  ... do stuff
  }

  if (exit_requested)
  {
    tsprintf(BOLDYELLOW "Termination signal received, closing.\n" RESET);
    for (int i = 0; i < n_students; i++)
    {
      char msg[MAX_MSG] = "exit message from manager\n";
      write(students[i].pipe_fd[1], msg, MAX_MSG);
      close(students[i].pipe_fd[1]);
    }
  }
```

8 Extras

Input coloring added via macros, this will work on Linux variants and macOS. If the program input is redirected to a file macros will printed to the file and look weird. It is designed to work on console only. Sample coloring below:



```
================================================
ragnaros@ubuntu:~/Desktop/POSIX-Threads$ sh execute.sh
================================================
5 students-for-hire threads have been created.
================================================
Name           Q      S      C
odtulu         5      3      900
bogazicili     4      5      1000
itulu          4      4      800
ytulu          3      4      650
sakaryali      1      2      200
================================================
odtulu is waiting for a homework.
bogazicili is waiting for a homework.
H has a new homework C; remaining money is 10000TL
ytulu is waiting for a homework.
itulu is waiting for a homework.
sakaryali is waiting for a homework.
sakaryali is solving homework C for 200, H has 9800TL left.
H has a new homework S; remaining money is 9800TL
bogazicili is solving homework S for 1000, H has 8800TL left.
H has a new homework Q; remaining money is 8800TL
odtulu is solving homework Q for 900, H has 7900TL left.
H has a new homework C; remaining money is 7900TL
ytulu is solving homework C for 650, H has 7250TL left.
H has a new homework S; remaining money is 7250TL
itulu is solving homework S for 800, H has 6450TL left.
H has a new homework Q; remaining money is 6450TL
bogazicili is solving homework Q for 1000, H has 5450TL left.
H has a new homework C; remaining money is 5450TL
ytulu is solving homework C for 650, H has 4800TL left.
H has a new homework S; remaining money is 4800TL
itulu is solving homework S for 800, H has 4000TL left.
H has a new homework Q; remaining money is 4000TL
bogazicili is solving homework Q for 1000, H has 3000TL left.
H has a new homework C; remaining money is 3000TL
odtulu is solving homework C for 900, H has 2100TL left.
H has a new homework S; remaining money is 2100TL
bogazicili is solving homework S for 1000, H has 1100TL left.
H has a new homework C; remaining money is 1100TL
sakaryali is solving homework C for 200, H has 900TL left.
H has no other homeworks, terminating.
No more homeworks left or coming in, closing.
ytulu is terminating.
itulu is terminating.
bogazicili is terminating.
odtulu is terminating.
sakaryali is terminating.
================================================
Name           Homeworks Solved    Money Made
odtulu         2                   1800
bogazicili     4                   4000
itulu          2                   1600
ytulu          2                   1300
sakaryali      2                   400
Total cost for 12 homeworks 9100TL
Money left at G's account: 900TL
================================================
ragnaros@ubuntu:~/Desktop/POSIX-Threads$
```