Algorithms Review for Job Interview

Jenny Huang

December 21, 2014

Contents

1 12/20/2014, Saturday 1

2 12/21/2014, Sunday 1

3 我是一个学生 1

1 12/20/2014, Saturday

- Website (github), program highlight, and chinese input environment all good now;
- Will configure Linux Mint Java environment later, prefer emacs;
- 145/168 done before new season review, begin to work on these questions from today.
- Just got 4 easiest done: **149/168**
  - min stack,
  - excel sheet column title,
  - compare version number, and
  - intersection of two linked list,

2 12/21/2014, Sunday

- Only three got done today: **152/169**
  - maximum gap
  - fraction to recurring decimal
  - majority element
- Don't feel my mind is clear today at all, will look into job searing instead, hopefully tomorrow I can solve more problems, and slightly complicated ones;

3 我是一个学生

```
#include <pthread.h>
#include <stdlib.h>
#include <stdio.h>

#define SIZE 8   // Size by SIZE matrices

using namespace std;

int main(int argc, char* argv[]) { // sampel mark for 中文是可以的
    pthread_t* thread;  // pointer to a group of threads
    int i;
```

```
    if (argc!=2) {
        printf("Usage: %s number_of_threads\n",argv[0]);
        exit(-1);
    }
    num_thrd = atoi(argv[1]);
    printf("num_thrd: %d\n", num_thrd);
    init_matrix(A);
    printf("\n");
    init_matrix(B);
    thread = (pthread_t*) malloc(num_thrd*sizeof(pthread_t));

    for (i = 1; i < num_thrd; i++) {
        //printf("address i: %d\n", i);
        int rc = pthread_create(&thread[i], NULL, multiply, &idx[i]);
        if (rc != 0) {
            perror("Can't create thread");
            free(thread);
            exit(-1);
        }
    }

    // main thread works on slice 0
    // so everybody is busy
    // main thread does everything if threadd number is specified as 1
    //int tmp = 0;
    multiply((void*)(&(idx[0])));

    // main thead waiting for other thread to complete
    for (i = 2; i <= num_thrd; i++)
        pthread_join(thread[i-1], NULL);

    printf("\n\n");
    print_matrix(A);
    printf("\n\n\t       * \n");
    print_matrix(B);
    printf("\n\n\t       = \n");
    print_matrix(C);
    printf("\n\n");

    free(thread);

    return 0;
}
```