#include <stdio.h>

#include <string.h>

#define TIME\_QUANTUM 4

#define NUM\_PROCESSES 4

typedef struct {

char name;

int burst\_time;

int remaining\_time;

int finished;

} Process;

int main() {

// Option D: P=3, Q=7, R=7, S=3

Process processes[NUM\_PROCESSES] = {

{'P', 3, 3, 0},

{'Q', 7, 7, 0},

{'R', 7, 7, 0},

{'S', 3, 3, 0}

};

int context\_switches = 0;

int s\_to\_q = 0, r\_to\_q = 0, q\_to\_r = 0, s\_to\_p = 0;

int time = 0;

int done = 0;

char prev = '\0';

while (done < NUM\_PROCESSES) {

for (int i = 0; i < NUM\_PROCESSES; i++) {

Process \*p = &processes[i];

if (p->remaining\_time > 0) {

// Context switch tracking

if (prev && prev != p->name) {

context\_switches++;

if (prev == 'S' && p->name == 'Q') s\_to\_q++;

if (prev == 'R' && p->name == 'Q') r\_to\_q++;

if (prev == 'Q' && p->name == 'R') q\_to\_r++;

if (prev == 'S' && p->name == 'P') s\_to\_p++;

}

printf("Time %d: Process %c runs\n", time, p->name);

int run\_time = (p->remaining\_time < TIME\_QUANTUM) ? p->remaining\_time : TIME\_QUANTUM;

p->remaining\_time -= run\_time;

time += run\_time;

prev = p->name;

if (p->remaining\_time == 0 && !p->finished) {

p->finished = 1;

done++;

printf("Process %c finished at time %d\n", p->name, time);

}

}

}

}

printf("\n=== Context Switches Summary ===\n");

printf("Total context switches: %d\n", context\_switches);

printf("S → Q: %d\n", s\_to\_q);

printf("R → Q: %d\n", r\_to\_q);

printf("Q → R: %d\n", q\_to\_r);

printf("S → P: %d\n", s\_to\_p);

printf("\n=== Constraint Check ===\n");

printf("S→Q == 1? %s\n", (s\_to\_q == 1 ? "YES" : "NO"));

printf("R→Q == 1? %s\n", (r\_to\_q == 1 ? "YES" : "NO"));

printf("Q→R == 2? %s\n", (q\_to\_r == 2 ? "YES" : "NO"));

printf("S→P == 0? %s\n", (s\_to\_p == 0 ? "YES" : "NO"));

return 0;

}