REPORT OS Assignment 4
Himangi Saraogi
201201074

Problem1: Life on a Chip

Problem Definition : A chip with n sites is given. There are a specified number of Hydrogen and Oxygen atoms. A reaction can take place at the sites to generate water molecules such that a reaction generates 1 unit of energy.We syncronize the formation of water molecules in an efficient and fast way.

Problem Constraints :

1) no two consecutive sites should be used for the reaction.
2) total energy generation at any moment should not exceed some threshold value.\

Assumptions:

1)After a reaction compleets in time 3 units the site is free and any other reaction can occur there immediately after.

Algorithm:

1)Find the maximum sites at which reaction can take place simultaneously max_sites = min(threshold_energy,(sites+1)/2).
2)Find number of water molecules that can be formed according to the limiting reagent.
3)For each molecule we create a thread that forms the molecules.
4)semaphores :

- a_block is used to syncronise decrement of left atoms.
- sites_block is used to synchronize access to sites array(1 where reaction is taking place).

5)In critical section we find a site with adjacent positions empty and such that the threshold energy has not been reached.The site is occupied and location is stored in tmp.
6)Then we decrement the Hydrogen and Oxygen atoms in critical section and wait 3 seconds for the reaction to take place.
7)Then we release the site in a critical section.
8)We join the thread to wait for thread completion .

Issues:

1) Since the thread creation also takes time it is possible that reactions occur at the same postions in a row and not at different positions each time.

Problem 2 GHAC Old Bridge Trip

Problem Definition: Given number of geeks and non-geeks. Synchronize crossing of the bridge by them.

Problem Constraints:
1)you cannot put three geeks with one non-geek or vice-versa.
2)if boarding group has a singer, then he can calm down even unbalanced group.
3)you can never allow two singers on same pass.

Assumptions:

1) There have to be exactly 4 people to cross the bridge.

Algorithm:
1)Each geek and non-geek has a separate thread which run the functions geekarrives() and nongeekarrives() respectively.
2) We use a single semaphore mutex to keep updation of waiting_geeks,waiting_nongeeks and singers left in critical section.
3)In geek arrives, after incrementing number of geeks in critical section
a)we check if wating geeks>=4, then we decrement waiting geeks by 4 and boardbridge() after leaving critical section.
b)waiting_geeks>=2 and waiting non_geeks>=2 : decrement geeks by 2 and non_geeks by 2 and boardbridge() after leaving critical section.
c)singers>0&&waiting_geeks==3 : decrement geeks by 3 and singers by 1 and boardbridge() after leaving critical section.
d)singers>0&& waiting_geeks=1 and waiting_non_geeks>=2 : decrement singers by 1, geeks by 1 and nongeeks by 2 and boardbridge() after leaving critical section.
e)else end the critical section and return.
4)If function has not returned till now then we call GoBridge as this means we have called BoardBridge exactly once.
5)We join the threads to wait for thread completion in main.

Problem 3 Courses Allocation Algorithm

Problem Definition: Allot given number of courses to given number of students.

Problem Constraints:
1)Each course allows upto 60 students to enroll.
2)Each course belongs to one of 4 group, also known as Knowledge Spectrum. For eg. Commerce, Humanities, Management, Arts etc.
3)Each course allows a certain quota to respective branch of students. For eg. M.Com, PHD, B.Com etc.
4)Each student belongs to one of four branch, i.e M.Com, PHD, B.Com and Arts.
5)Each student can take exactly 4 courses in one semester.
6)Each student is allowed to make a list of preferences to the courses they like, such that size of list is 8.
7)Each student must pick at least one course from each Knowledge spectrum, i.e Commerce,Arts etc.
8)Each student should get 4 courses from the prefernce of 8.
9)Quota for each branch is in ratio 1:1:2:1 i.e out of 60 studnets, 12 belong to M.com, 12 to PHD, 24 to B.com and 12 to Arts.

Assumptions:
1) The courses such that Course_number%4 is 0,1,2,3 are in Knowledge spectrum Commerce, Humanities, Management, Arts respectively.
2) The preference list and the branch of a student are randomly generated.

Algorithm:
1)Initialize student and course data structures depending on number of students and courses.
2)for each student create a thread that runs the allot function.
3)In the allot function, we separate out the preference function into 4 queues of courses belonging to different groups in the Knoledge spectrum.
4)If any queue is empty, we return as allocation is not possible.
5)semaphore: mutex[i] to prevent the structure of same course from being modified simultaneously.
6)for all the 4 queues:
• set a flag to 0.
• for each course in the queue check if there are vacancies for the students branch.
• If yes,allot course to student and break. If no continue.
7)If no course could be alloted in any of the queues we exit as allocation is not possible.
8)We join the threads to wait for thread completion in main.