

AM:1115201800009 ΤΥΠΑΛΔΟΣ-ΠΑΥΛΟΣ ΑΠΟΣΤΟΛΑΤΟΣ

το πρόγραμμα ακολουθεί μια πολιτική που λέει ότι ένας saladmaker όσο είναι “ξυπνιος” φτιάχνει σαλάτες.

Για να τρέξετε την εργασία καλέστε να 4 εκτελεσιμα από 4 terminals με το ./saladmaker να έχει flag -y που να του επιδικνύει πιο υλικο(n k ‘η p) έχει πάντα στην κατοχή του. Παραδειγμα

```
./saladmaker -t1 0.2 -t2 0.4 -s 294913 -y k
```

το shmid θα σας το εκτυπώσει η ./chef

πιστευω τα σχολια εξηγουν τα προγραμματιστικα ζητηματα και την νοοτροπια του προγραμματος.

η μεταφορα υλικων και η ενημερωση των σαλατοποιων οτι χρωστανε σαλατες να γινεται μέσω του value του semaphore είναι πολυ πιο εξυπνο απο το memory segment να είναι το τραπέζι οπου υπάρχουν χ1 ντοματες χ2 κρεμμυδια και χ3 πιπεριες. Θα γινονταν δυσνοητο δυσπρογραμματιστο. Εξαλλου ετσι οι semaphores δε θα παιζουν με values μονο 0 και -1 με wait and sleep να είναι wake up και sleep buttons και είναι πιο πολυ στα μετρα της εκφωνησης που λέει "Ο διευθυντής 'ειδοποιεί' τον σαλατοποιό να πάρει τα υλικά που χρειάζεται".

Επισης για αυξηση της ταχυτητας εβαλα το value των semaphores να είναι ο αριθμος των σαλατων που χρωστουν -1 αρα οι σαλατοποιοι να μην κοιμουνται όσο εχουν υλικά για και αλλες σαλατες αλλα οι βοηθοι στο piazza ειπαν ο σεφ να μην δινει τα ιδια υλικά 2 φορες αρα μέσω του shared memory εβαλα boolean που να δειχνουν ποτε ένας σεφ δεν εχει υλικά για να στέλνει σε αυτον μονο τοτε.

Προβληματα:οπως θα δειτε στο ενιαιο logfile το append γραψιμο απετυχε να είναι παραλληλο με την δραστηριοτητα ενος εκτελεσιμου να γραφεται ολοκληρη μετα την αντιστοιχη ενος αλλου εκτελεσιμου. Παρολα αυτα οι χρονοι που αναγραφονται είναι πραγματικοι και ετσι κανεις μπορει να δει ποτε καθε διεργασία εκανε τι

για να δειτε οτι ολες οι σαλατες φτιαχτηκαν καντε ενα ctrl+f στο txt και ψαξτε End making salad για να δειτε οτι το βρισκει οσες φορες όσο γραψατε στο -n του ./chef

