

# README

Ξανθοπούλου Μαρία , AM : 1115201400300

## 1. Παραδοτέο Αρχείο

Το αρχείο μορφής <.tar> περιέχει τα αρχεία: Makefile, το βασικό πρόγραμμα askisi3.c, το header file header.h, ένα αρχείο κειμένου atext.txt και το παρών αρχείο README.pdf

## 2. Εκτέλεση Προγράμματος

Το πρόγραμμα αφού μεταγλωτιστεί με την εντολή “make” ,μπορεί να εκτελεστεί μέσω του εκτελέσιμου αρχείου “output” που δημιουργήθηκε. Η εντολή έχει τη εξής μορφή “./output <N> <K> ” όπου “N” είναι ο αριθμός των διεργασιών , “K” είναι το πλήθος των επαναλήψεων (μηνυμάτων που στέλνονται από τα C προς το P).

## 3. Σχεδιασμός Προγράμματος

Το πρόγραμμα ξεκινάει παίρνοντας απο την γραμμη εντολής 2 παραμετρους N και K όπως προαναφέρθηκε. Δημιουργεί N διεργασίες P (Producer) , και μια parent process C (Consumer). Οι διεργασίες αυτές μπορούν να επικοινωνούν μέσω της κοινής μνήμης που δημιουργείται. Προκειμένου το πρόγραμμα να μπορεί να τρέξει με παραπάνω απο έναν producer θα πρέπει να χρησιμοποιήσουμε system V semaphores. Έτσι έχοντας 6 σεμαφόρους ( 2 mutex , 2 για τον producer και 2 για τον consumer) ξεκινάει η υλοποίηση του ζητούμενου.

Κάθε διεργασία P έχει πρόσβαση στο αρχείο κειμένου atext.txt απο όπου και επιλέγει μια τυχαία γραμμή. Κάνουμε down στον σημαφόρο WP ( WriteProducer) και το κείμενο που επιλέγει μαζί με το pid του αποθηκεύονται στο struct in , το οποίο στέλνουμε στην κοινή μνήμη. Τότε κάνουμε up τον σημαφόρο RC ( ReadConsumer) ώστε να μπορεί τότε ο Consumer να διαβάσει το struct in απο την κοινή μνήμη και ταυτόχρονα κάνουμε up στον σημαφόρο WP ( WriteProducer) για να μπορέσει να λάβει νέο μήνυμα. Ο consumer διαβάζει το struct in παίρνει το τυχαίο string και κάνει md5 κωδικοποίηση. Τότε κάνει down στον σημαφόρο WC (WriteConsumer) , γράφει στο struct out το νέο κωδικοποιημένο string και το pid του Producer , και τα στέλνει στην κοινή μνήμη . Τότε κανουμε up στον σημαφόρο RP ( ReadProducer) ο οποίος διαβάζει το struct out απο τη μνήμη και αν το pid του struct out είναι ίδιο με το pid του Producer τότε αυξανουμε τη μεταβλητη pid\_match. Τέλος , κάνουμε up στον σημαφόρο WC( WriteConsumer) ο οποίος θα γράψει το νέο μήνυμα στο struct out.

Αυτό γίνεται για K επαναληψεις και όταν το K εξαντληθεί , ο Consumer ειδοποιεί τα child processes ( P ) μέσω ενός handler για να σταματήσουν.

Στο τέλος εκτυπώνονται τα στατιστικά του προγράμματος.

Για την υλοποίηση του Md5 απαιτείται η εγκατασταση της ssl-dev lib.