**Λειτουργικά Συστήματα Υπολογιστών**

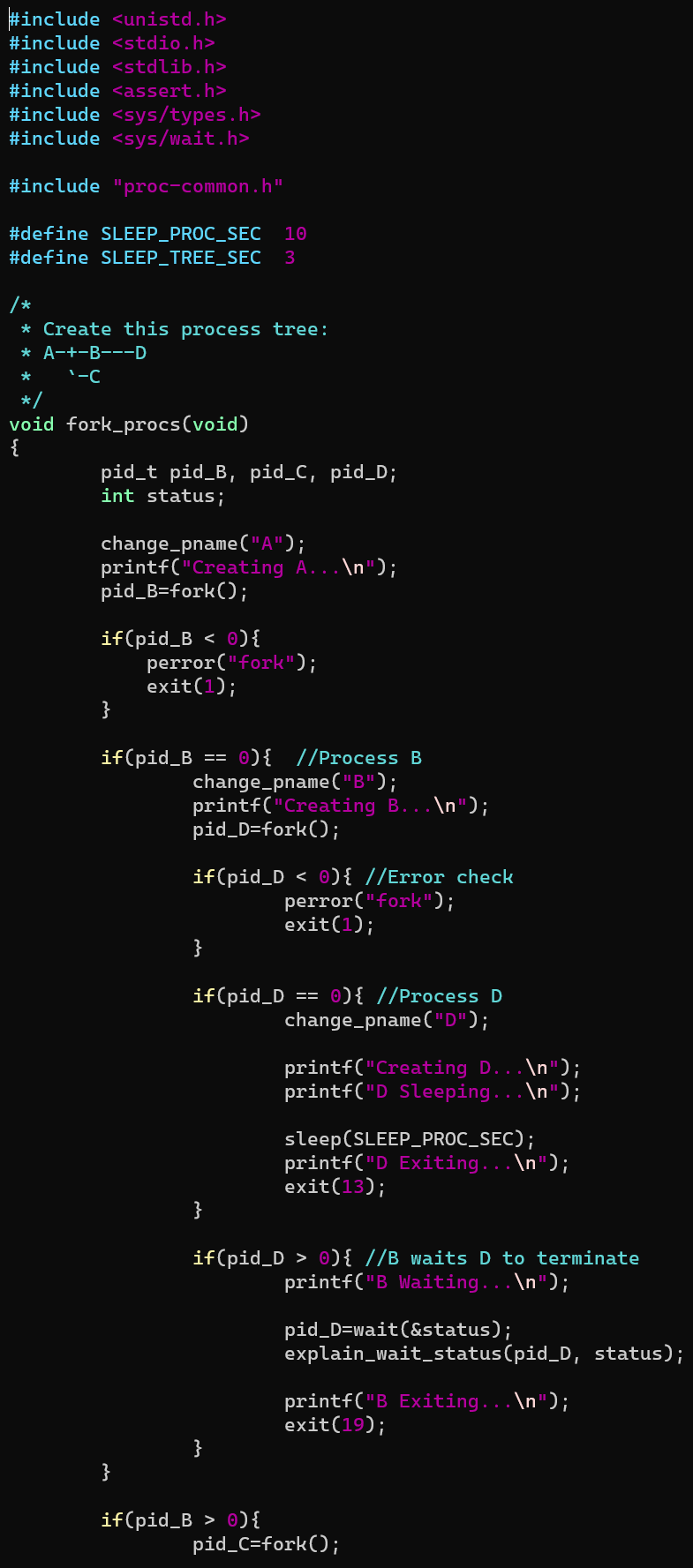
*Αναφορά στην 2η Εργαστηριακή Άσκηση*

Αλέξανδρος Σκούρας, 03120105

Ιωάννης Τσαντήλας, 03120883

*Εξάμηνο: Εαρινό 2022-23*

* 1. **Δημιουργία δεδομένου δέντρου διεργασιών**



Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Η έξοδος του προγράμματος:

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

**Σημείωση**

Η συνάρτηση **wait()**είναι ένα system call που αναστέλλει την λειτουργία μιας διεργασίας έως ότου κάποιο από τα παιδιά της τερματιστεί. Όταν συμβεί αυτό, το λειτουργικό στέλνει το κατάλληλο σήμα ώστε ο πατέρας να συνεχίσει (για αυτό, εάν ο πατέρας έχει πολλαπλά παιδιά, καλούμε την **wait**πολλαπλές φορές).

**Ερωτήσεις**

1. Εάν σκοτώσουμε πρόωρα την διαδικασία Α με **kill -KILL <pid>**, τότε όλα τα παιδιά-διαδικασίες της Α (δηλαδή οι B,C) θα γίνουν ορφανές διαδικασίες, οι οποίες θα αποκτήσουν ως νέο πατέρα την init διαδικασία, με PID=1.
2. Εάν γράψουμε **show\_pstree(getpid())** αντί για **show\_pstree(pid)**η έξοδος του προγράμματος είναι:

Εικόνα που περιέχει διάγραμμα, σχηματικό

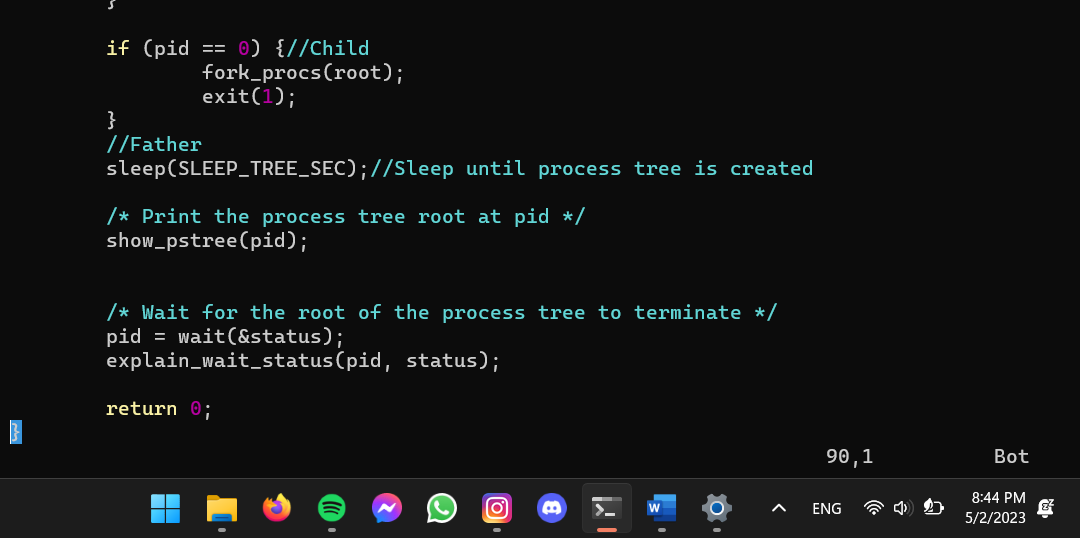
Περιγραφή που δημιουργήθηκε αυτόματα

Όπου**test1**είναι το εκτελέσιμο του προγράμματος, **sh**και **pstree**είναι διαδικασίες πουπροκύπτουν από την **show\_pstree()**. Με την χρήση του **pid**, καλούμε την συνάρτηση για την διαδικασία Α, ενώ με την **getpid()**την καλούμε για το εκτελέσιμο, εξ’ ου και οι επιπλέον διαδικασίες.

1. Αυτό μπορεί να συμβαίνει επειδή οι **διαδικασίες καταναλώνουν πόρους** του συστήματος, όπως CPU time, μνήμη και δίσκο Ε/Ε, επομένως πολλές διαδικασίες μπορούν να μειώσουν την αποδοτικότητα του συστήματος. Άλλοι λόγοι είναι η **ασφάλεια** περιορίζοντας το πλήθος των πόρων που οι κακόβουλοι hackers θα μπορέσουν να αξιοποιήσουν και **δικαιοσύνη**, ώστε κάθε χρήστης να έχει ισάξιο πλήθος πόρων.
   1. **Δημιουργία αυθαίρετου δέντρου διαδικασιών**

**Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα**

****

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματαΓια το εξής αρχείο εισόδου:

Έχουμε έξοδο:

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματαΕικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

**Ερωτήσεις**

1. Τα μηνύματα «εισόδου» (starting) τυπώνονται με βάση την BFS διάσχιση του δέντρου(Βέβαια και εδώ μπορεί να υπάρξει τυχαιότητα). Τα μηνύματα «εξόδου» (exiting) δεν έχουν την ίδια σειρά με αυτά της εισόδου ή με την σειρά που εκτελούνται. Η σειρά εξαρτάται από τον αλγόριθμο δρομολόγησης του λειτουργικού, τον εκτελούμενο χρόνο κάθε διαδικασίας ή το πλήθος των διαδικασιών που συναγωνίζονται για τους πόρους. Επιπλέον, εάν ο πατέρας περιμένει πολλά παιδιά να τερματιστούν, η σειρά τερματισμού δεν είναι απαραίτητα προκαθορισμένη.
   1. **Αποστολή και χειρισμός σημάτων**

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Για το ίδιο αρχείο εισόδου με το ερώτημα 1.2, έχουμε έξοδο:

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

**Ερωτήσεις**

1. Τα πλεονεκτήματα της χρήσης σημάτων ενάντια της **sleep()** είναι η **αμεσότητα** και την **ακρίβεια** που προσφέρουν στον χειρισμό των διαδικασιών, όπου σε αντίθεση με την **sleep()**, μπορούμε να ελέγξουμε ακριβώς πότε θα σταματήσει/συνεχίσει, προσφέροντας μας **ασύγχρονη λειτουργία** του προγράμματος μας.
2. Η **wait\_for\_ready\_children** περιμένει όλα τα παιδιά της διεργασίας που την κάλεσε να σταματήσουν. Την χρειαζόμαστε έτσι ώστε να εξασφαλίσουμε πως κάθε παιδί-διεργασία έχει δημιουργηθεί και είναι έτοιμο να εκτελεσθεί πριν από τον πατέρα-διεργασία, διαφορετικά θα προκύψουν προβλήματα συγχρονισμού, όπως π.χ. η αλληλεπίδραση του πατέρα με ένα μη έτοιμο παιδί.
   1. **Παράλληλος υπολογισμός αριθμητικής έκφρασης**

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Για τα παρακάτω αρχεία εισόδου, έχουμε αντίστοιχα:

Εικόνα που περιέχει κείμενο, συσκευή

Περιγραφή που δημιουργήθηκε αυτόματα Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

**Ερωτήσεις**

1. Στην συγκεκριμένη περίπτωση, και επειδή οι πράξεις της **πρόσθεσης** και του **πολλαπλασιασμού** είναι αντιμεταθετικές, χρειαζόμαστε και χρησιμοποιούμε **μία** μόνο σωλήνωση για την επικοινωνία της κάθε διεργασίας γονέα με τα παιδιά του(Δεν μας ενδιαφέρει η σειρά επιστροφής των αποτελεσμάτων στον γονέα). Σε αντίθετη περίπτωση, στις πράξεις της **αφαίρεσης**, **ακέραιας διαίρεσης** και **ακέραιου υπόλοιπου**, οι οποίες δεν είναι αντιμεταθετικές, θα χρειαστούμε **δύο** σωληνώσεις για την επικοινωνία της κάθε διεργασίας γονέα με τα παιδιά του(Μας ενδιαφέρει η σειρά επιστροφής των αποτελεσμάτων στον γονέα).
2. Το βασικό πλεονέκτημα είναι η απόδοση. Υπολογίζοντας παράλληλα πολλαπλές διεργασίες, ο συνολικός χρόνος εκτέλεσης είναι μικρότερος από αυτόν που θα χρειαζόμασταν εάν υπολογίζαμε κάθε διεργασία ξεχωριστά.