

עבודה להגשה מס' 4

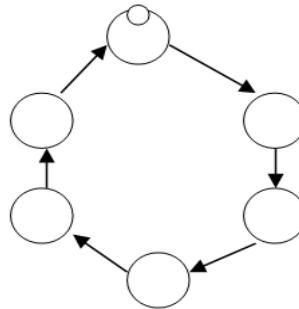
~Processes and Pipes~

תאריך הגשה: 13/05/21 בשעה 23:55

- קראו היטב את השאלות.
- ניתן להגיש עבודה בזוגות.
- הגשת העבודה תהיה דרך אתר הקורס במודל.
- בעמוד הראשון של העבודה יש לרשום את שמות ומספרי ת.ז. של המגישים.
- יש להגיש את העבודה בקובץ zip בתוכו יהיו קבצים:
Ex4.c readme.txt
- **שם הקובץ שיוגש למערכת ההגשה יהיה מורכב מת"ז של המגיש/ים. לדוגמה:**
111111111.zip עבור הגשה ביחיד
111111111_222222222.zip עבור הגשה בזוג
- במקרה של הגשה בזוגות, רק אחד מבני הזוג יגיש את העבודה במודל.
- איחור במועד ההגשה יגרור הורדה של ציון, 5 נק' לכל יום איחור או חלק ממנו.
בכל מקרה לא יהיה ניתן להגיש מעבר לשלושה ימי איחור ממועד ההגשה המקורי.
במקרים חריגים בלבד יש לפנות למרצה כדי לקבל אישור על הגשה באיחור.
- שאלות לגבי העבודה יש לשאול בפורום באתר הקורס ("מודל") או בשעות קבלה של המתרגל/ת האחראי/ת **בלבד** (מיכל צ'וברוצקי). **אין לשלוח שאלות במייל המתרגלים או המרצה.**

עבודה נעימה!

בהינתן n תהליכים נרצה לבנות אלגוריתם מתייצב עצמית אשר בכל רגע נתון מאפשר לכתוב אך ורק לתהליך אחד על המסך. דבר ראשון נסדר את התהליכים במעגל:



נבחר מנהיג (תהליך ראשי) ונבצעה אלגוריתם הבא:

עבור תהליך מנהיג:

1. קרא ערך של התהליך שלפניך (האחרון במעגל)
2. אם הערך של התהליך זהה לערך שלך אזי
 - א. יש עדכן את הערך של המנהיג ל- $(\text{ערך} + 1) \pmod{n}$
 - ב. יש להדפיס את הערך של המנהיג על המסך

עבור תהליך שאינו מנהיג:

1. קרא ערך של התהליך שלפניך
2. אם הערך שונה משלך אזי
 - א. יש עדכן את הערך של התהליך לערך של התהליך הקודם
 - ב. יש להדפיס את הערך של התהליך על המסך

כתבו תוכנית אשר תקלוט מהמשתמש מספר שלם חיובי, תיצור מעגל של n תהליכים, לכל אחד מהתהליכים יוגרל ערך התחלתי שרירותי וינתן תפריט הבא למשתמש:

- | | |
|---|---|
| 0 | הצג ערכים של התהליכים |
| 0 | בצע צעד (כל תהליך מבצע צעד אחד בדיוק "מערכת סינכרונית") |
| 0 | יציאה |

לתשומת ליבכם- אם מימשתם נכון את האלגוריתם, אזי לאחר מספר צעדים בכל צעד תודפס בדיוק הודעה אחת ובמעגל יהיו לכל היותר 2 מספרים שונים עוקבים.

ראו דוגמה לפלט בעמוד הבא.

```
Please enter number of processes to create: 6
```

```
-----
```

```
Menu
```

- 1. Show values
- 2. Pulse
- 0. Exit

```
>>1
```

```
The process 5 has value 5.
The process 4 has value 4.
The process 3 has value 3.
The process 2 has value 2.
The process 1 has value 1.
The process 0 has value 0.
```

```
-----
```

```
Menu
```

- 1. Show values
- 2. Pulse
- 0. Exit

```
>>2
```

```
Permission to write at process 4
Permission to write at process 5
Permission to write at process 3
Permission to write at process 2
Permission to write at process 1
```

```
-----
```

```
Menu
```

- 1. Show values
- 2. Pulse
- 0. Exit

```
>>0
```

עבודה נעימה!