**Operating systems Ultra code exam 2022 A' מבחן במערכות הפעלה אולטרא קוד 2022 א**

**הנחיות המבחן:**

**יש לקרוא את ההנחיות בעיון רב**

**כללי התנהלות בזמן הבחינה:**

**חומר עזר מותר/אסור:**

* שימוש במחשבון **מותר** .
* חומר העזר **המותר**, כמפורט בפלטפורמה: הדפסה של סיכום הקור ס באנגלית/עברית, הדפסה של תקציר סיכוםהקורס באנגלית/עברית, הדפסה של המצגות של הגב׳ נחמה היימן.
* השימוש בטלפונים סלולריים, מילונים אלקטרוניים **אסו ר** .
* חומר הקורס בפלטפורמה הקורס **אסו ר** לשימוש.
* גישה לחומר לימוד מקוון, במחשב, ברשת או בשרת – **אסורה**.
* המבחן נערך תחת מערכת safe-exam. ניסיון לצאת מהמבחן ולעבור לאתר אינטרנט אחר, או לחומר בפלטפורמה, בזמן המבחן, יביא להגשה אוטומטית של המבחן, ולאי יכולת של הנבחנת לחזור ולהמשיך את פתרון המבחן.
* העברת חומר עזר מתלמידה אחת לרעותה, **אסורה בהחלט**. שימוש בחומר עזר משותף יגרום לפסילת הבחינות של כל השותפים בו.

**מהלך המבחן:**

* יש לשמור על דממה מוחלטת בשעת הבחינה.
* משך זמן המבחן הוא **3 שעות** .
* המבחן הוא מבחן עם שאלות רב ברירתיות )״אמריקאיות״(. בחלק מהשאלות, את מתבקשת לנמק את בחירתך.

ניתן לנמק בעברית.

* עליך לענות על כל שאלות המבחן )אין שאלות בחירה(.
* חלק מהתמודדות עם החומר הנלמד, היא הבנה של השאלות הנשאלות. לכן לא תהיה אפשרות של שאילת שאלות במהלך המבחן. כל המידע הנדרש כדי לענות תשובה נכונה מופיע בגוף השאלה.
* כתשובה לכל שאלה עליך לסמן את הבחירה הנכונה ביותר.
* אם נתקלת בבעיה טכנית במהלך המבחן, פני מיד לצוות והן תעזורנה למצוא פתרון לבעיה.

נותר לנו רק לאחל לך בהצלחה במבחן

צוות הקור ס

**שאלה 1) 5 נק׳(**

Which of the following actions can be performed only in kernel mode?

? kernel mode איזו פעולה מבין הפעולות הבאות ניתנת לביצוע אך ורק במצב ה

1. Performing I/O operations directly (without the operating system’s help).

ביצוע פעולות קלט-פלט באופן ישיר

1. thread-switch when it comes to user threads.

.user-threads כאשר מדובר בthread-switch

1. Assigning a value into a global variable. .הצבת ערך במשתנה גלובאלי

1. All of the other operations described in all the other options should be enabled only in kernel mode.

את כל הפעולות האחרות הנתונות בסעיפים האחרים יש לאפשר אך ורק בkernel mode .

1. Change of system settings that require an administrator password.

. שינוי הגדרות המערכת שדורשות סיסמת מנהל

**שאלה 2) 5 נק׳(**

Which of the following conditions ensures the prevention of a deadlock in a system with four different types of RESOURCES: : שוניםRESOURCES במערכת עם 4 סוגיdeadlock מה מבין התנאים הבאים מבטיח מניעת

1. Each process will request no more than 2 RESOURCES.

כלprocess יוכל לבקש לא יותר משניRESOURCES .

1. Each process will release **all the** RESOURCES it holds (including synchronization means) before it ends.

כלprocess ישחרר את **כל ה-**RESOURCES שהוא מחזיק בהם )כולל אמצעי סנכרון( לפני שיסתיים.

1. No process will have a higher priority than any other process. . אחרprocess לא תהיה עדיפות גבוהה יותר מprocess לשום

1. After a process has received **all the** RESOURCES it has requested, it will release them **all** within a maximum of 10 seconds.

אחרי ש-process קיבל את **כל**RESOURCES שביקש, הוא ישחרר את **כולם** בתוך 10 שניות לכל היותר.

1. No answer from the given answers is correct. . אף תשובה מבין הנתונות איננה נכונה

**שאלות 8-3 )15 נק׳: 5 נק׳ כל אחד(**

**כל שאלה מחולקת ל-2:**

**שאלה רב ברירתית שלאחריה מקום לנמק את הבחירה שלך .**

Select the best synchronization primitive for each synchronization problem to solve the problem. You are not required to implement a complete algorithm. The algorithm can include data structures/variables necessary for the solution of the problem, but you do not need to deal with them. It is possible to use the primitive as many times as required (except in questions that state precisely differently). You don't have to worry about how you will access the synchronization primitive; you should assume that you have access to it.

בשביל כל אחת מבעיות ה synchronization שבהמשך, בחרי את אמצעי ה synchronization הטוב )כלומר היעיל(

ביותר בכדי לפתור את הבעיה. אין צורך לממש אלגוריתם שלם. באלגוריתם יכולים להיות מבני נתונים/משתנים

שהכרחיים לפתרון, אין צורך להתייחס אליהם. אפשר להשתמש בכל אמצעי כמה פעמים שנדרש )חוץ מן המקריםשבתשובה אפשרית מופיע במפורש אחרת(. אין צורך לדאוג כיצד מתאפשרת הגישה לאמצעי הsynchronization , ניתן להניח שיש גישה.

**שאלה 3) 5 נק׳(**

A team of programmers is developing an appointment scheduling system at an Emergency Medical Center. The scheduling system should allow each person to perform the following operations in parallel:

* Registration for a free appointment.
* Cancellation of a previously registered appointment.

Because this is an emergency medical center that deals with urgent cases, everything that connects to the system is treated as an individual THREAD. If there are no available appointments, the THREAD goes into standby mode. Still, the customer does not disconnect from the system- he continues to wait until an appointment is available.

In the first stage of development, it was decided to implement a system without considering FAIRNESS issues (FAIRNESS: the user that tried first would be first to register for an appointment). It was agreed that the first stage would be implemented, the implementation’s performance would be checked, and only after that would the development continue.

Can you offer a synchronization primitive that will allow an effective solution to the requirements of the first stage ?

צוות מתכנתים מפתח מערכת זימון תורים במרכז לרפואה דחופה שאמורה לאפשר לכל המעוניינים ביצוע מקבילי של

הפעולות הבאות:

* הרשמה לתור פנוי.
* ביטול הרשמה.

בגלל שמדובר במקרים דחופים, כל אדם המתחבר למערכת מטופל כ THREAD נפרד ואם אין תור פנוי, הTHREAD עובר למצב המתנה והלקוח לא מתנתק מן המערכת אלא מחכה עד שיתפנה התור.

בשלב הראשון הוחלט לממש מערכת בלי לדאוג להוגנותFAIRNESS ( FAIRNESS : מי שינסה ראשון יוכל להירשםראשון (. הוחלט לבדוק את תפקודה לאחר פיתוח השלב הראשון ורק אחרי זה להמשיך בפיתוח בשלבים הבאים .

האם את יכולה להציע אמצעי synchronization שיאפשר פתרון יעיל לדרישות של השלב הראשון?

1. A Counting semaphore
2. A Binary semaphore
3. A Lock (without a CV)
4. A Lock and a Condition variable with a Condition signal
5. A Lock and a Condition variable with a Condition broadcast

**שאלה 4:**

: explain your answer for question 3 – 3 נמקי תשובתך לשאלה

הלקוח נכנס להמתנה על התנאי וכשהתנאי יתקיים – ז"א כשיתפנה תור-הלקוח יקבל הודעה – וינסה לתפוס אותו.

זהו בדיוק מצב של המתנה על cv כשמתפנה תור – מתקיים תנאי נשלח סיגנל להעיר את הממתינים.

המיוטקס אינו הוגן.

מיותר להשתמש בbroadcast היות והתור מתפנה רק ללקוח בודד.

**שאלה 5) 5 נק׳(**

As part of the development of a traffic simulation software, it is necessary to simulate the operation of a traffic light on a multi-lane road. Each car is implemented as a separate THREAD. In the first stage of development, it is necessary to enable two operations:

* Stop all cars at a traffic light when its color changes to red.
* Simultaneous starting of movement of all cars when the traffic lights’ color changes to green. There is no need to handle SYNCHRONIZATION between various cars during traffic (among themselves – this is implemented by a different team).

A team of programmers is examining various synchronization constructs that will allow the most straightforward and most efficient solution to this problem. Suggestions?

כחלק מפיתוח תוכנת הדמיית תנועה בכביש ,יש צורך לדמות פעולת רמזור בכביש רב-נתיבי. כל מכונית ממומשת כ

THREAD נפרד. בשלב הראשון של הפיתוח יש צורך לאפשר 2 פעולות:

* עצירת כל המכוניות ברמזור ,כאשר הצבע שלו מתחלף לאדום.
* התחלה בו זמנית של תנועת כל המכוניות כשהצבע הרמזור מתחלף לירוק. אין צורך לטפל ב SYNCHRONIZATION בין מכוניות בזמן התנועה )בינם לבין עצמם, זה תפקיד של צוות אחר(.

צוות מתכנתים בוחן אמצעי synchronization שונים שיאפשרו את הפתרון הפשוט והיעיל ביותר לבעיה. הצעות?

1. A Counting semaphore
2. A Binary semaphore
3. A Lock (without a CV(
4. A Lock and a Condition variable with a Condition signal
5. A Lock and a Condition variable with a Condition broadcast

**שאלה 6:**

:explain your answer for question 5 – 5 נמקי תשובתך לשאלה

הרמזור קורא לכל המכוניות להמשיך בנסיעה.

**שאלה 7) 5 נק׳(**

In a computer game, there is an overloading problem on a two-way bridge. Each car travels during the computer game and is represented by an individual THREAD. It has been decided that an intelligent student like you can easily solve the two-way bridge overloading problem with the help of synchronization primitives - that will ensure that at any point in time, there will be no more than two cars on the bridge (in total in both directions). The primitive/primitives must create the most efficient solution possible. Which of the following primitive/primitives would you suggest using?

במשחק מחשב קיימת בעיית עומס יתר בגשר דו-כיווני. המכוניות נוסעות במהלך משחק המחשב ,כשכל מכונית היא THREAD נפרד. הוחלט שתלמידה חכמה כמוך, תוכל לפתור בקלות את הבעיה בעזרת אמצעי synchronizationשיבטיח/יבטיחו שבכל נקודת זמן לא יהיו יותר מ 2 מכוניות על הגשר(בשני הכיוונים). האמצעי/אמצעים חייב/חייביםלהיות היעיל ביותר. מה את מציעה?

1) Counting semaphore

2) 2 Binary semaphores

3) 2 Counting semaphores

4( 2 MUTEX (LOCKs)

5) A Lock and a Condition variable with a Condition signal

**שאלה 8:**

:explain your answer for question 7 – 7 נמקי תשובתך לשאלה

סמפור מונה מאפשר גישה ליותר ממשאב אחד, במקרה זה 2 מכוניות יכולות לעלות לגשר, ולכן נאתחל את ערך המונה ל-2 כך שתתבצע המתנה רק לאחר ש2 מכוניות על הגשר.

אין צורך ב2 מנעולים היות ואין הגבלה לצד מסוים של הגשר אלא הגשר באופן כללי מאפשר עד 2 מכוניות.

**שאלה 9) 10 נק׳(**

You are given the following program (in pseudo-code), which implements the SYNCHRONIZATION between several PROCESSES running in parallel. Each PROCESS executes the same CRITICAL SECTION entry and exit protocol.

The program uses N BINARY SEMAPHORES initialized to 1, and Si is the name of the ith SEMAPHORE (S1, S2, ... SN). N is a natural number greater than 2.

נתונה התוכנית הבאה(pseudo code) שמיועדת לSYNCHRONIZATION בין מספר כלשהו שלPROCESSES שרצים במקביל. כלPROCESS מבצע אותו פרוטוקול כניסה ויציאה לCRITICAL SECTION .

התוכנית משתמשת בBINARY SEMAPHORES ,N שאותחלו ל 1, כאשרSi  הוא שם שלi SEMAPHORE . כלומר, 1S2 , S,

... עדN .SN  הוא מספר טבעי גדול מ- 2.

|  |
| --- |
| Protocol 1 |
| while (1){ for (i=0; i++; i<N) wait(Si);  /\* Critical section \*/ for (i=N-1; i--; i>=0) post (Si);  } |

Choose the correct claim:

בחרי את הטענה הנכונה:

1. Two processes can be in the CRITICAL SECTION at the same time.

שני תהליכים יכולים לשהות בו-זמנית בCRITICAL SECTION .

1. Two processes may enter a DEADLOCK state.

שני תהליכים עלולים להיכנס למצבDEADLOCK .

1. The protocol ensures the precedence of a process created first over the other competing processes. .הפרוטוקול מבטיח קדימות של תהליך שנוצר ראשון על פני התהליכים המתחרים

1. The protocol ensures the precedence of a process created last over the other competing processes. .הפרוטוקול מבטיח קדימות של תהליך שנוצר אחרון על פני התהליכים המתחרים

1. The protocol solves the CRITICAL SECTION problem.

הפרוטוקול פותר את בעיית הCRITICAL SECTION .

**שאלה 10:**

:explain your answer to question 9 – 9 נמקי תשובתך לשאלה

**שאלה 11 )5 נק׳(**

You are given the following program (in pseudocode), which is an implementation of the SYNCHRONIZATION between several PROCESSES running in parallel (the program is similar to the program in the previous question - Protocol 1, but the SEMAPHORES are released, after the end of the CRITICAL SECTION, in ascending order instead of descending order). Each PROCESS executes the same CRITICAL SECTION entry and exit protocol.

The program uses N BINARY SEMAPHORES initialized to 1, and Si is the name of the ith SEMAPHORE (S1, S2, ... SN). N is a natural number greater than 2.

נתונה התוכנית הבאה(pseudo code) שמיועדת לSYBCHRONIZATION בין מספר כלשהו שלPROCESSES שרצים במקביל )התוכנית דומה לסעיף הקודם עם פרוטוקול 1, אבל הSEMAPHORES משתחררים אחרי סיום הCRITICAL SECTION בסדר עולה במקום יורד(. כלPROCESS מבצע אותו פרוטוקול כניסה ויציאה לCRITICAL SECTION .

התוכנית משתמש ת בBINARY SEMAPHORES ,N שאותחלו ל 1, כאשרSi  הוא שם שלi SEMAPHORE . כלומר, 1S2 , S,

... עדN .SN  הוא מספר טבעי גדול מ- 2.

|  |
| --- |
| Protocol 2 |
| while (1){ for (i=0; i++; i<N) wait (Si);  /\* Critical section \*/ for (i=0; i++; i<N) post (Si);  } |

Choose the correct claim:

בחרו את הטענה הנכונה:

1. Two processes can be in the CRITICAL SECTION at the same time. . CRITICAL SECTION שני תהליכים יכולים לשהות בו-זמנית ב

1. Two processes may enter a DEADLOCK state. .DEADLOCK שני תהליכים עלולים להיכנס למצב

1. The protocol ensures the precedence of a process created first over the other competing processes. .הפרוטוקול מבטיח קדימות של תהליך שנוצר ראשון על פני התהליכים המתחרים

1. The protocol ensures the precedence of a process created last over the other competing processes. .הפרוטוקול מבטיח קדימות של תהליך שנוצר אחרון על פני התהליכים המתחרים

1. The protocol solves the CRITICAL SECTION problem. . CRITICAL SECTION הפרוטוקול פותר את בעיית ה

**שאלה 12:**

:explain your answer to question 11 – 11 נמקי תשובתך לשאלה

**13 )10 נק׳(**

Assuming that a particular run did not enter a deadlock state and two or more PROCESSES were not in a critical segment at the same time, select the correct statement in terms of the effectiveness of protocols

1 and 2 (without addressing the question of whether it is a correct solution to a critical section problem).

בהנחה שבהרצה מסוימת לא נוצר קיפאון ושניPROCESSES או יותר לא שהו בו זמנית בקטע קריטי, בחרו בטענה הנכונה מבחינת היעילות של הפרוטוקולים 1 ו 2 )מבלי להתייחס לשאלה האם הוא מהווה פתרון נכון לבעיית קטע קריטי(.

1. The two protocols are equally efficient.

שני הפרוטוקולים יעילים באותה המידה.

1. Protocol 1 is more efficient because it reduces the number of WAKEUPS and SLEEPS of processes.

פרוטוקול 1 יעיל יותר בגלל שהוא חוסך במספר הערות WAKEUPS והרדמותSLEEPS של תהליכים.

1. Protocol 2 is more efficient because it reduces the number of WAKEUPS and SLEEPS of processes.

פרוטוקול 2 יעיל יותר בגלל שהוא חוסך במספר הערות WAKEUPS והרדמותSLEEPS של תהליכים.

1. Protocol 2 is more efficient because it releases SEMAPHORES in the same order they were caught.

פרוטוקול 2 יעיל יותר בגלל שהוא משחררSEMAPHORES לפי אותו סדר שהם נתפסו.

1. Protocol 1 is more efficient because it first releases the SEMAPHORE that all other processes are waiting for. . שכל התהליכים האחרים מחכים לוSEMAPHORE פרוטוקול 1 יעיל יותר בגלל שהוא קודם כל משחרר את ה

**שאלה 14:**

:explain your answer to question 13 – 13 נמקי תשובתך לשאלה

**שאלה 15 )10 נק׳(**

The file system of a particular operating system uses the I-node method with the following parameters :

* + The BLOCK size of the file system is 1 Kbyte
  + The BLOCK address on the disk is 4 bytes

•The 12 I-node fields can directly hold the DATA BLOCK address on the disk

* + One additional field is intended to hold the address of the single indirect block
  + One more field is intended to hold the address of the double indirect block
  + And one more field is intended to hold the address of the triple indirect block

The size of a particular file in the system is 1000 Kbyte. What is the total number of BLOCKS required to hold this file in the file system (including the pointer BLOCKS, but not including the block that contains the i-node of the file)?

מערכת הקבצים של מערכת הפעלה מסוימת משתמשת בשיטת הI-node עם פרמטרים הבאים:

* + גודל הBLOCK במערכת הקבצים הוא 1Kbyte .
  + כתובת הBLOCK בדיסק היא 4 בתים) bytes(.
  + 12 שדות של ה- I-node יכולים להחזיק ישירות את כתובת הDATA BLOCK בדיס ק
  + שדה נוסף אחד נועד להחזיק כתובת של ה- single indirect block
  + עוד שדה נוסף אחד נועד להחזיק כתובת של ה- double indirect block
  + ועוד שדה נוסף אחד נועד להחזיק כתובת של ה-triple indirect block

גודלו של קובץ מסוים במערכת הוא 1000Kbyte .

מהי כמות הBLOCKS ,בסה"כ שדרושה להחזקת קובץ זה במערכת הקבצים )כולל גם אתBLOCKS של מצביעים, אבל לא כולל את הבלוק שמכיל את ה-i-node של הקובץ(?

* 1. 1000
  2. 1003
  3. 1005
  4. 1010
  5. 1011

**16**

What is true about VMM (virtual machine monitor) in bare metal configuration?

? bare metal בתצורת )virtual machine monitor( VMM מה נכון לגבי

* + 1. All hardware resources are directly controlled by the hypervisor.

. hypervisor נשלטים **באופן ישיר** ע"יhardware resources כל משאבי החומרה

* + 1. Only main memory allocations and I/O operations go through the hypervisor, but the physical CPU processor time allocation is performed **directly** by the guest OS **without** **the** **participation** of the hypervisor.

רק הקצאות זיכרון ראשי ופעולות קלט-פלט עוברות דרךhypervisor , אבל הקצאת זמן המעבדCPU הפיזי

מבוצעת **ישירות** ע"י הOS האורחת **בלי שיתוף** של hypervisor .

* + 1. Only main memory allocations pass through the hypervisor, but I/O operations are performed **directly** by the guest OS **without the participation** of the hypervisor.

האורחתOS אבל פעולות קלט-פלט מבוצעות **ישירות** ע"י ה ,hypervisor רק הקצאות זיכרון ראשי עוברות דרך

**בלי שיתוף** של hypervisor .

* + 1. Because an operating system must be installed on **at least one virtual machine**, it is possible to run **user applications directly on the VMM** without having to install any **additional** operating systems within the other virtual machines. בגלל שחייבת להיות מערכת ההפעלה מותקנת **בלפחות מכונה וירטואלית אחת**, יש אפשרות הרצה של **יישומי** . בלי צורך בהתקנת מערכות הפעלה **נוספות** כלשהן בתוך שאר המכונות הווירטואליות**VMM משתמש הישר על ה**

* + 1. When the user application makes the system calls to the OS that running on VMM, the same OS will **not attempt** to make the system calls in kernel mode. **לא תנסה** לבצע את OS אותה ה ,VMM שרצה עלOS- לsystem calls כאשר יישום המשתמש מבצע את

. kernel mode בsystem calls -ה

**17**

The page table of a process in a system with virtual memory looks like this. All numbers are decimal numbers, starting from zero, and all addresses are addresses of bytes in memory. Page size is 1024 bytes.

טבלת הדפים של תהליך במערכת עם זיכרון וירטואלי נראית כך. כל המספרים הם דצימליים, מתחילים מאפס, וכל

הכתובות הן כתובות של בייטbyte בזיכרון. גודל הדף הוא 1024 בייטים.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Page Number | Valid bit | Frame Number |
|  | 0 | 1 | 1 |
|  | 1 | 1 | 2 |
|  | 2 | 0 | 1 |
|  | 3 | 1 | 1 |
|  | 4 | 0 | 3 |
|  | 5 | 1 | 0 |

To which physical address, if any, will virtual address 1054 be mapped? ?1054 לאיזה כתובת פיזית, אם יש כזו, תמופה כתובת וירטואלית

00100 0001 1110=

2\*1024+30=2078

2078 )1

8222 )2

# 1054 )3 1078 )4

5150 )5

**18**

Are there any errors in the page table from question 17 and if so, what are they? Please mark the most accurate answer.

האם יש שגיאות בטבלת הדפים משאלה 17 ואם כן, מה הן? נא לסמן את התשובה המדויקת ביותר.

1. No errors. .אין שגיאות

1. There is an error - Page 5 is in frame 0 whose number is lower than the number of the frame in which page 0 is found.

יש שגיאה - דף 5 נמצא במסגרת 0 שמספרה נמוך יותר ממספר המסגרת שדף 0 נמצא בה.

1. There is only one error - page 0 and page 3 are present in the frame with the same number, this is not proper.

יש רק שגיאה אחת - דף 0 ודף 3 נמצאים )נוכחים( במסגרת עם אותו מספר, הדבר אינו תקין.

באותה טבלת דפים אין אפשרות להצביע לאותה מסגרת וגם דף 0 וגם דף 3 מסמונים כתקינים valid

1. There is only one error - page 2 and page 3 are present in the frame with the same number, this is not proper.

יש רק שגיאה אחת - דף 2 ודף 3 נמצאים )נוכחים( במסגרת עם אותו מספר, הדבר אינו תקין.

1. There are 2 errors - pages 0, 2, and 3 are present in a frame with the same number, this is not proper.

יש 2 שגיאות - גם דף 0 וגם דף 2 וגם דף 3 נמצאים )נוכחים( במסגרת עם אותו מספר, הדבר אינו תקין.

**שאלה 19 )10 נק׳(**

A system that runs on a CPU with **8 cores** where each core can run one THREAD, handles many tasks simultaneously and is built in the form of a boss-workers model with the transfer of tasks through the queue. The tasks (and also the boss) are only computational (99% CPU bounded). The boss runs with a much higher priority than workers. To improve the systems’ speed, a very large main memory is installed so no virtual memory is needed. The tests showed that the average time to perform one task (time from start of execution to completion) by a worker is 2.2 seconds. The time to handle a new task by the boss from the moment the request is received until it is entered into the task queue is 0.5 seconds. The task arrival rate is 2 per second. How many workers is it recommended to create so that the task handling time (waiting + execution to completion) is the smallest?

מערכת שרצה על המעבד **עם 8 ליבות** )CPU with 8 cores( כשכל ליבה יכולה להריץTHREAD אחד, מטפלת במקביל בהרבה משימות ובנויה בצורתboss-workers model עם העברת המשימות דרך התור. המשימות )וגם הboss ( הן חישוביות בלבד )CPU bounded 99%(. הboss רץ עם עדיפות גבוהה בהרבה מworkers . כדי לשפר את מהירות

המערכת, הותקן זיכרון ראשי גדול מאוד ולכן אין צורך בזיכרון וירטואלי. הבדיקות הראו שזמן הממוצע לביצוע המשימה

אחת )זמן מהתחלת הביצוע עד לסיומו( ע"יworker הוא 2.2 שניות. זמן טיפול במשימה חדשה ע"יboss מרגע קבלת הבקשה עד להכנסתה לתור המשימות הוא 0.5 שנייה. קצב הגעת המשימות הוא 2 בשנייה. כמהworkers מומלץ ליצור כדי שזמן טיפול במשימה )המתנה + ביצוע עד הסיום( יהיה הקטן ביותר?

# 16 )1 5 )2 4 )3 7 )4 8 )5

(קצב ההגעה הוא 2 משימות לשניה ולכן אחרי 5 משימות הוורקר הראשון כבר מתפנה)

**שאלה 20 )5 נק׳(**

What will be the correct answer to question 19 when it comes to a CPU with 4 cores - how many workers are recommended to create, in the new situation, so that the task handling time (standby + execution to completion) will be the smallest?

מה תהיה התשובה הנכונה על שאלה 19 כשמדובר על המעבד **עם 4 ליבות** )CPU with 4 cores( - כמהworkers מומלץ ליצור ,במצב החדש ,כדי שזמן טיפול במשימה )המתנה + ביצוע עד הסיום( יהיה הקטן ביותר?

# 8 )1 4 )2 3 )3 5 )4 2 )5

**21 )5 נק׳(**

What is true about comparing information transfer between PROCESSES via SHARED MEMORY or through MESSAGE PASSING using an operating system messaging service?

מה נכון לגבי השוואת בין העברת מידע בין הPROCESS –ים דרךSHARED MEMORY או דרךMESSAGE PASSING בעזרת שירות הודעות של מערכת הפעלה?

1. Transferring information by multiple MESSAGE PASSINGS is faster than transferring the same amount of data through shared memory.

מהירה יותר מהעברת אותה כמות הנתונים דרך זיכרוןMESSAGE PASSINGS

העברת מידע ע"י מספר רב של . משותף

1. The 2 options have the same speed because both are executed by the operating system.

ל 2 הצורות אותה מהירות בגלל ששתיהן מבוצעות ע"י מערכת ההפעלה.

1. It is not advisable to create a SHARED MEMORY area for data transfer if there is only one transfer in total. . להעברת הנתונים אם מדובר בהעברה אחת בלבד בסה"כSHARED MEMORY לא כדאי ליצור איזור

1. MESSAGE PASSING is very effective only between different THREADS and not between different PROCESSES.

MESSAGE PASSING יעילה ביותר רק בין הTHREADS שונים ולא ביןPROCESSES שונים.

1. Data transfer via SHARED MEMORY is **especially** suitable for different THREADS of the same process. . שונים של אותו תהליךTHREADS מתאימה **במיוחד** לSHARRED MEMORY העברת הנתונים דרך

**שאלה 22 )5 נק׳(**

Goggle Docs is a platform that allows you to edit WORD documents without installing software and that enables the user to work with these documents on their local computer. What kind of cloud services should be associated with Google Docs?

לאיזה סוג שירותי ענן נכון לשייך "מסמכי גוגל"Goggle Docs )פלטפורמה שמאפשרת לערוך מסמכי WORD בלי להתקין

במחשב לוקאלי תוכנה שמסוגלת לעבוד עם המסמכים האלה(?

1. Infrastructure as a Service (IaaS) because a user is unaware of the amount of physical resources he is consuming.

Infrastructure as a Service (IaaS) בגלל שמשתמש לא מודע לכמות המשאבים הפיזיים שהוא תופס.

1. Platform as a Service (PaaS) because editing is done on a different platform from the local computer.

Platform as a Service (PaaS) בגלל שעריכה מתבצעת בפלטפורמה שונה מהמחשב הלוקאלי.

1. Software as a Service (SaaS) because the service is provided for free.ֿ

. בגלל שהשירות ניתן בחינםSoftware as a Service (SaaS)

1. Platform as a Service (PaaS) because the service includes an operating system.

. בגלל ששירות כולל מערכת הפעלהPlatform as a Service (PaaS)

1. Software as a Service (SaaS) because the service includes ready-made applications in addition to hardware and an operating system.

Software as a Service (SaaS) בגלל שהשירות כולל אפליקציות מוכנות בנוסף על חומרה ומערכת הפעלה.