מדינת ישראל
משרד החינוך

סוג הבחינה: גמר לבתי־ספר לטכנאים ולהנדסאים
מועד הבחינה: אביב תשפ"ו, 2026
סמל השאלון: 714913

נספחים:
א. דף תשובות
ב. נספח פקודות לחלק א'
ג. טופולוגיה לחלק ב'
ד. נספח פקודות לחלק ב'
ה. מילון מונחים

עקרונות מערכות הפעלה, תקשורת נתונים ורשתות מחשבים

הוראות לנבחנים

א. משך הבחינה: ארבע שעות.

ב. מבנה השאלון ומפתח ההערכה: בשאלון זה שני חלקים.

חלק א': עקרונות מערכות הפעלה	50	נקודות
חלק ב': תקשורת נתונים ורשתות מחשבים	50	נקודות
סך־הכול	100	נקודות

ג. חומר עזר מותר בשימוש: מחשבון פשוט שאינו ניתן לתכנות.

ד. הוראות מיוחדות:

1. ענו על השאלות **אך ורק** על גבי דף התשובות שבנספח א'.
2. יש להחזיר למשגיחים **רק** את דף התשובות (נספח א').
3. לשאלון זה מצורף מילון מונחים בשפות עברית, ערבית, אנגלית ורוסית. תוכלו להיעזר בו בעת הצורך.

הוראות למשגיחים:
בתום הבחינה, יש לוודא שהנבחנים הדביקו את מדבקת הנבחן שלהם
במקום המיועד לכך בדף התשובות שבנספח א', וצירפו אותו למחברת הבחינה.

כתבו **במחברת הבחינה בלבד**, בעמודים נפרדים, כל מה שברצונכם לכתוב **כטיוטה** (ראשי פרקים, חישובים וכדומה).
כתבו "טיוטה" בראש כל עמוד טיוטה. כתיבת טיוטות כלשהן על דפים שמחוץ למחברת הבחינה עלולה לגרום לפסילת הבחינה!

בשאלון זה 33 עמודים ו־25 עמודי נספחים.

השאלות בשאלון זה מנוסחות בלשון רבים,
אף על פי כן על כל תלמידה וכל תלמיד להשיב עליהן באופן אישי.

בהצלחה!

המשך מעבר לדף ◀

השאלות

חלק א': עקרונות מערכות הפעלה (50 נקודות)

שימו לב: את כל התשובות יש לסמן אך ורק על גבי דף התשובות שבנספח א'.
בשאלות רב־ברירה, לכל שאלה ארבע תשובות, אך רק אחת מהן נכונה. הקיפו את הספרה המציינת את התשובה הנכונה. בשאלות האחרות, פעלו על־פי ההנחיות.

פרק ראשון (30 נקודות)

ענו על כל השאלות 1–15 (לכל שאלה – 2 נקודות).

שאלה 1

נתון קטע קוד של תוכנית.

```
#include <stdio.h>
#include <unistd.h>
#include <string.h>
#include <errno.h>

int main() {
    if (symlink("../dir1", "./dir1/link_to_dir2") == -1) {
        perror("Error creating symlink");
    }
    return 0;
}
```

א. מה יהיה הפלט של התוכנית?

1. הקוד יצליח, וייווצר קישור סימבולי חוקי.
2. הקוד יצליח, אך הקישור שייווצר יגרום ללולאה אין־סופית.
3. הקוד ייכשל, מכיוון שקישור סימבולי לא יכול להצביע על ספרייה.
4. הקוד ייכשל, וייווצר קישור שבור, כי הוא מצביע על נתיב יחסי.

ב. מדוע יצירת hard link לספרייה אסור בחלק ממערכות הקבצים?

1. כי קישור קשיח שייווצר יפנה רק ל־inode של הספרייה ולא לתוכן שלה.
2. כי מבנה ה־inode של ספרייה לא תומך בריבוי קישורים אליו.
3. כי קישור קשיח חייב להצביע על קובץ רגיל ולא על ספרייה.
4. כי הדבר עלול ליצור מעגלים (loops) בספריית הקבצים, והדבר יקשה על מערכת ההפעלה לבצע חיפושים רקורסיביים.

המשך בעמוד 3

שאלה 2

מה מכילה טבלת הפסיקות (interrupt vector table) בעת ריצת תוכנית כלשהי במחשב?

1. הפניות לשגרות טיפול בפסיקות שנקבעו על ידי ה־BIOS.
2. שגרות טיפול בפסיקות שנקבעו על ידי מערכת ההפעלה.
3. הפניות לשגרות טיפול בפסיקות כפי שנקבעו בפעם האחרונה שהתבצע revectoring.
4. שגרות טיפול בפסיקות כפי שנקבעו על ידי המשתמש.

שאלה 3

מה יקרה כתוצאה מהרצת השורה "chmod -R 755 /home/student/test" לתיקייה test?

1. מצב ההרשאות ישתנה לפי הטבלה הבאה, כולל הודעות שגיאה.

	User	group	other
read	V	V	V
write	V	V	V
execute	V	X	X

2. מצב ההרשאות ישתנה לפי הטבלה הבאה לכל הפריטים בתיקייה.

	User	group	other
read	V	V	V
write	V	V	V
execute	V	X	X

3. מצב ההרשאות ישתנה לפי הטבלה הבאה, כולל הודעות שגיאה.

	User	group	other
read	V	V	V
write	V	X	X
execute	V	V	V

4. מצב ההרשאות ישתנה לפי הטבלה הבאה לכל הפריטים בתיקייה.

	User	group	other
read	V	V	V
write	V	X	X
execute	V	V	V

המשך בעמוד 4

שאלה 4

לפניכם סקריפט שהורץ במערכת לינוקס על ה־shell, על מנת לחשב את נפח כל הקבצים בתיקייה מסוימת.

```
1.    #!/bin/bash
2.    directory="/path/to/your/directory"
3.    total_size=$(du -csh "$directory" | grep total | awk '{print $1}')
4.    echo "Total size of files in $directory: $total_size"
```

מה יש לשנות בקוד כדי להפעיל אותו על התיקייה הנמצאת במיקום: /Ubuntu/home/student/test

1. לשנות את שורה 1 ל: #!/bin/shell
2. לשנות את שורה 2 ל: directory="/home/student/test"
3. לשנות את שורה 2 ל: directory="/Ubuntu/home/student/test"
4. לשנות את שורה 3 ל: total_size=$(du -csh "directory" | grep total | awk '{print $1}')

שאלה 5

לפניכם סקריפט:

```
host="$1"
if [ -z "$host" ]; then
echo "Usage: $0 <hostname or IP>"
exit 1
fi
ping -c 4 "$host"

if [ $? -eq 0 ]; then
echo "$host is reachable."
else
echo "$host is not reachable."
fi
```

מה מבצע הסקריפט?

1. בודק חוקיות כתובת IP נתונה.
2. מוודא חיבור לאתר מסוים.
3. מנסה להתחבר לרשת.
4. כל התשובות נכונות.

המשך בעמוד 5

שאלה 6

תהליכים מתוזמנים למעבד בשיטת Round Robin, Time Slice(q) = 15ms, להלן טבלת התהליכים:

מזהה תהליך	עדיפות	זמן ב־ms
A	2	13
B	1	15
C	4	17
D	3	20

כמה פרצי מעבד צריך כדי שכל התהליכים יסיימו לרוץ בהצלחה?

1. ארבעה
2. חמישה
3. שישה
4. שבעה

שאלה 7

איזה מהבאים הוא ההיבט הקריטי ביותר שיש לקחת בחשבון בעת שימוש ב־Thread-safe data structures?

1. ביצועים
2. קלות התחזוקה
3. שימוש בזיכרון
4. היכולת להשתמש בהם גם בתוכניות חד־תהליכיות

שאלה 8

איזה מהבאים <u>אינו יתרון</u> של Paging בהשוואה לניהול זיכרון רציף?

1. מבטל לחלוטין את הפיצול הפנימי (Internal Fragmentation).
2. מאפשר שימוש יעיל יותר בזיכרון הפיזי על ידי מניעת פיצול חיצוני (External Fragmentation).
3. מאפשר מימוש של זיכרון וירטואלי.
4. מספק הפרדה טובה יותר בין מרחבי כתובות של תהליכים שונים.

המשך בעמוד 6

שאלה 9

נתונה תוכנית שצריכה לבצע חישובים מתמטיים מורכבים על אלפי תמונות, ולאחר מכן לכתוב את התוצאות לקובץ יחיד. אילוץ מרכזי הוא שהתוכנית צריכה להישאר רספונסיבית למשתמש בזמן החישובים, ולא "להיתקע". איזו גישה של שימוש בתהליכונים מתאימה ביותר לפתרון בעיה זו, ומדוע?

1. שימוש בתהליכון ראשי עבור ממשק המשתמש, ויצירת מספר מוגבל של תהליכוני עבודה (worker threads) שיבצעו את החישובים במקביל וישמרו את התוצאות במבנה נתונים משותף, כאשר תהליכון נפרד ידאג לאיסוף התוצאות ולכתיבתן לקובץ.
2. יצירת תהליכון נפרד עבור כל תמונה כדי לבצע את החישובים, ותהליכון נפרד נוסף לכתיבת כל התוצאות לקובץ.
3. שימוש בתהליכון ראשי יחיד לכל המשימות, כולל החישובים והכתיבה לקובץ.
4. יצירת תהליכון רקע אחד שיבצע את כל החישובים באופן סדרתי, ורק לאחר מכן את הכתיבה לקובץ.

שאלה 10

מדוע יומן journaling (קבצי לוג) במערכת קבצים נחשב לשיפור משמעותי?

1. כי הוא מקטין את כמות המקום הפנוי על הדיסק.
2. כי הוא מאפשר שחזור מהיר של מערכת הקבצים לאחר קריסה.
3. כי הוא מונע גישה לא מורשית לקבצים.
4. כי הוא מאפשר גישה מהירה יותר לקבצים קטנים.

שאלה 11

מה היתרון בשימוש בזיכרון וירטואלי (virtual memory)?

1. מאפשר לתהליכים להשתמש בשטח זיכרון כולל גדול יותר ממה שמותקן פיזית על המחשב.
2. מגדיל את קיבולת הזיכרון הפיזי של מערכת.
3. מבטל את הצורך בניהול זיכרון.
4. משפר את הביצועים של כל יישומי המערכת.

שאלה 12

נתון מחשב שהזיכרון הראשי בו מנוהל בשיטת סיגמנטים. חומרת המחשב תומכת בארבעה סיגמנטים. נתונה תוכנית הכתובה בשפת c ושמורה בקובץ exe.c, ובהרצתה נוצר תהליך p.

מי אחראי על חלוקת מרחב הכתובות הלוגי של התהליך לסיגמנטים?

1. מנהל המערכת
2. מערכת ההפעלה
3. חומרת הדיסק
4. מתכנת הקובץ exe.c

המשך בעמוד 7

שאלה 13

מה ההבדל בין צינור (pipe) ל־FIFO?

1. צינור משמש לתקשורת בין תהליכים, ו־FIFO משמש לתקשורת בין חוטים (threads).
2. צינור הוא חד־כיווני, ו־FIFO הוא דו־כיווני.
3. צינורות יעילים יותר מ־FIFOs.
4. אין הבדל; התנאים ניתנים להחלפה.

שאלה 14

מה המטרה של טבלת דפים בניהול זיכרון וירטואלי (virtual memory management)?

1. לאחסן את תוכן הזיכרון הפיזי
2. לנהל את הקצאת הזיכרון הפיזי
3. מיפוי כתובות וירטואליות לכתובות פיזיות
4. לטפל בפעולות קלט/פלט של קבצים

שאלה 15

בתהליך המשתמש בזכרון Paging, מה תהיה הפעולה של המעבד כאשר מתרחש 'Page Fault'?

1. מעביר את התוכנית למצב 'Running' וממשיך לבצע את ההוראה.
2. ממשיך בריצה תוך כדי טעינת העמוד החסר לזיכרון הראשי.
3. מעביר את התהליך למצב 'Blocked' ושולח פסיקה למערכת ההפעלה.
4. מסיים את התהליך ומעביר אותו למצב 'Terminated'.

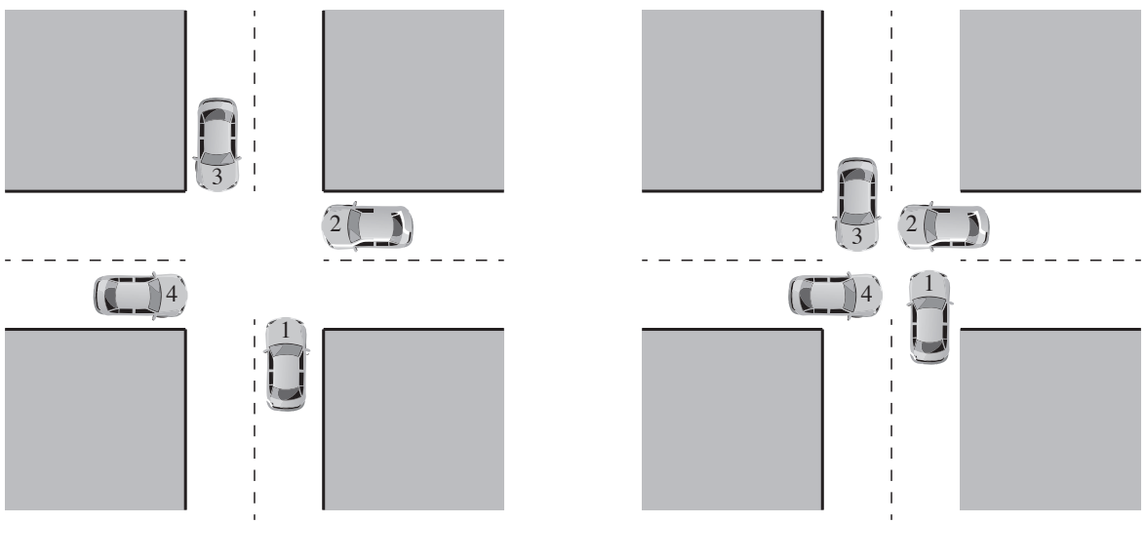
המשך בעמוד 8

<u>פרק שני</u> (20 נקודות)

ענו על <u>כל</u> השאלות 16-25 (לכל שאלה – 2 נקודות).

שאלה 16

ארבע מכוניות נכנסות לצומת, כמתואר בתרשים a. אם כל המכוניות ייכנסו בו זמנית לצומת, המערכת תגיע לחסימה הדדית, כמתואר בתרשים b.



(a) Deadlock possible **(b) Deadlock**

איור לשאלה 16

בחרו את הפעולה שתמנע את החסימה מלהתרחש.

1. קביעת סדר עדיפויות בין המכוניות בכניסה לצומת. (מכונית תוכל להיכנס רק על פי העדיפות שלה).
2. מכונית תיכנס לצומת רק אם אין עוד מכוניות ממתינות בכניסות אחרות לצומת.
3. כל מכונית תעצור לאחר כל מטר של נסיעה, כדי לוודא שהיא לא חוסמת שום מכונית אחרת.
4. כל התשובות נכונות.

המשך בעמוד 9

שאלה 17

עליכם לשקול פתרון לבעיית הפילוסופים האוכלים, המשלב שני מנגנונים:

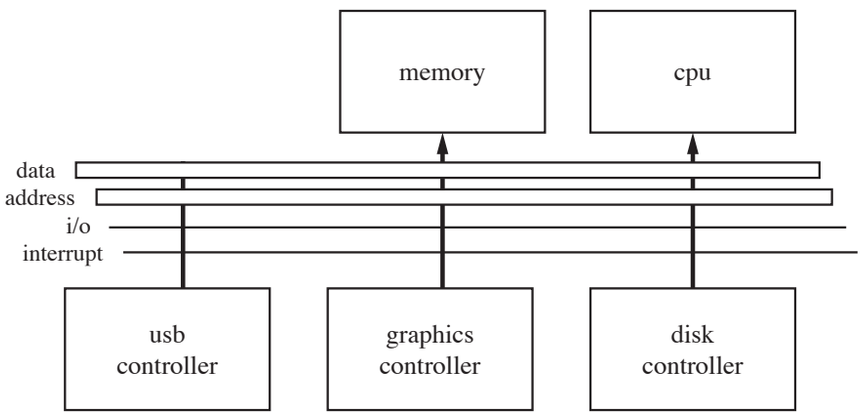
- כל פילוסוף מורשה לקחת מזלג רק אם שני המזלגות שלו זמינים בו־זמנית.
- פילוסופים אי־זוגיים לוקחים קודם את המזלג השמאלי ואז את הימני, ופילוסופים זוגיים לוקחים קודם את המזלג הימני ואז את השמאלי.

איזו טענה נכונה לגבי פתרון משולב זה?

1. הפתרון מונע 'Deadlock' על ידי מניעת תנאי 'Hold and Wait' וגם 'Circular Wait'.
2. הפתרון גורם ל־'Livelock' בגלל ריבוי התנאים.
3. הפתרון מונע 'Deadlock' אך עלול לגרום ל־'Starvation' לפילוסופים מסוימים.
4. הפתרון פשוט וקל ליישום אך אינו מבטיח עקביות בנתונים.

שאלה 18

במחשב מסויים בנוסף לכונן הבזק (USB) ודיסק קשיח (R–disk) יש גם שימוש במעבד גרפי חזק. התרשים הבא מתאר את המערכת.



איור לשאלה 18

כיצד אפשר לייעל את המערכת?

1. לנהל תור עדיפויות
2. להוסיף בקר למעבד
3. לשנות את רוחב הפס
4. להשתמש במספר קווי פסיקה בפס

המשך בעמוד 10

שאלה 19

תהליך A רץ על ליבת מעבד בודדת. במהלך ביצועו, הוא מנסה לגשת לכתובת וירטואלית שאינה טעונה כרגע בזיכרון הראשי (RAM) – אירוע המכונה Page Fault. ה־RAM מלא, וקיים תהליך B במצב 'Ready' בעדיפות גבוהה יותר. מהו רצף האירועים הסביר ביותר שיקרה מהרגע שבו המעבד זיהה את ה־Page Fault ועד שתהליך B מתחיל לרוץ, ומה ההשלכה על תהליך A?

1. המעבד מטפל מייד בפסיקה, טוען את הדף החסר מהדיסק ל־RAM, וממשיך את ריצת תהליך A ללא החלפת הקשר.
2. המעבד מעלה פסיקת חומרה, והמערכת מבצעת מייד החלפת תהליך (Context Switch) לתהליך B, מבלי להשעות את תהליך A. הדף של תהליך A נטען אוטומטית מהדיסק במקביל, ויוצר 'Livelock'.
3. המעבד מעלה פסיקת חומרה ומעביר את השליטה למערכת ההפעלה, המעבירה את תהליך A למצב 'Blocked-Suspended' תוך כדי בחירת דף להחלפה. תהליך B נבחר על ידי המתזמן ומתחיל לרוץ, בעוד הדף של תהליך A נטען מהדיסק, וגורם להשהיה גבוהה עבור תהליך A.
4. המעבד מזהה את ה־Page Fault ומעביר את תהליך A למצב 'Ready', המתזמן בוחר את תהליך B ומריץ אותו, וההמתנה לטעינת הדף מתרחשת במקביל.

שאלה 20

נניח כי קיימת מערכת הפעלה עם שני משאבים בלעדיים (שאינם ניתנים לשיתוף), R1 ו־R2, המוגנים על ידי Semaphores.
תהליך A מחזיק ב־R1 וממתין להשגת R2.
במקביל, תהליךB מחזיק ב־R2 וממתין להשגת R1.
מהי התוצאה הסבירה ביותר של מצב זה, ומהם מנגנוני הליבה שייכנסו לפעולה כדי לטפל בו?

1. יתרחש 'Race Condition', והתהליך שיגיע ראשון למשאב השני יקבל אותו.
2. יתרחש 'Starvation', והמתזמן ייתן לתהליך אחד עדיפות על פני השני.
3. התהליכים ייחסמו במצב 'Deadlock' עקב המתנה מעגלית, ומערכת ההפעלה תשתמש במנגנון 'גילוי והתאוששות' (Deadlock Detection and Recovery).
4. התהליכים יעברו למצב 'Livelock' וינסו שוב ושוב להשיג את המשאבים, מבלי להתקדם, עד להתרחשות פסיקה.

המשך בעמוד 11

שאלה 21

מערכת הפעלה מודרנית משתמשת בתזמון 'Round Robin' עם חלון זמן (Time Slice) קצר ביותר, ובניהול זיכרון מבוסס Paging.
תהליך A מבצע פעולות קלט/פלט אינטנסיביות, ותהליך B מבצע חישובים אינטנסיביים (CPU-bound). שניהם רצים במקביל על מעבד בודד.
מהי התוצאה הסבירה ביותר של אינטראקציית הביצועים ביניהם, וכיצד היא קשורה למנגנוני הזיכרון והתזמון?

1. שני התהליכים יסבלו מ־'Thrashing' עקב חלונות זמן קצרים ו־Context Switches תכופים, והדבר יפגע ביעילות זיכרון המטמון ויוביל לריבוי Page Faults.
2. תהליך A יסבול מ־'Starvation' בגלל הקצאת זמן CPU מועדפת לתהליך B, ותהליך B ייהנה מ־Hit Rate גבוה בזיכרון מטמון.
3. מערכת ההפעלה תזהה אוטומטית את סוג התהליך ותקצה לו חלון זמן ארוך יותר כדי למנוע Page Faults.
4. תהליך B יסיים את ריצתו במהירות יחסית מכיוון שהוא CPU–bound, ותהליך A ימתין לרוב ל־I/O ויפנה את ה־CPU ביעילות.

שאלה 22

נתונה מערכת מרובת ליבות המריצה תהליך יחיד (Single-Threaded) עם זמן ריצה של 100ms, ותהליך מרובה-תהליכונים (Multi-Threaded) עם 4 תהליכונים (Threads) שכל אחד מהם דורש זמן ריצה של 30 ms .
המתזמן (Scheduler) הוא מסוג SJF (Shortest Job First) שאינו דורס (non-preemptive).
כיצד תתנהג המערכת, ומדוע?

1. התהליך היחיד והתהליכונים (Threads) ירוצו במקביל, כיוון שהמערכת מרובת ליבות. כתוצאה מכך, זמני ההמתנה של שניהם יהיו קצרים מאוד.
2. תתרחש 'Starvation', מכיוון שהתהליך היחיד יקבל עדיפות ויידחוק את התהליכונים (Threads).
3. התהליך היחיד ירוץ תחילה עד לסיומו, ולאחר מכן ארבעת התהליכונים (Threads) ירוצו עד לסיומם, כיוון שהם 'jobs' קצרים יותר. זמני ההמתנה של התהליכונים (Threads) יהיו ארוכים יותר.
4. התהליך היחיד ירוץ תחילה עד לסיומו, ולאחר מכן ארבעת התהליכונים (Threads) ירוצו עד לסיומם, כיוון שלתהליך היחיד יש עדיפות גבוהה יותר.

שאלה 23

נתונה מערכת מרובת מעבדים (Multi-Core) המריצה תוכנית שכוללת שלושה תהליכים המסומנים כ־T1, T2, ו־T3.

המתזמן (Scheduler) הוא מסוג FCFS (First Come First Served) שאינו דורס (non-preemptive).
זמני הריצה של כל תהליך הם: T1 = 10ms, T2 = 5ms, T3 = 12ms.
בהיתנן שסדר הגעת התהליכים הוא: t1-> t2-> t3 (משמאל לימין)

מהו סדר הריצה של התהליכים ומה יהיה זמן ההמתנה הממוצע שלהם? (יש לקרוא את סדר התהליכים משמאל לימין).

1. T1, T2, T3; זמן המתנה ממוצע: 9.66 ms
2. T1, T2, T3; זמן המתנה ממוצע: 8.33 ms
3. T2, T1, T3; זמן המתנה ממוצע: 11.67 ms
4. T1, T2, T3; זמן המתנה ממוצע: 14.33 ms

המשך בעמוד 12

שאלה 24

נתונה מערכת מרובת ליבות (Multi-Core) המריצה תהליך (P1) עם שלושה תהליכונים (T1, T2, T3) על ליבה אחת.

תהליך נוסף (P2) נמצא במצב 'Running' על ליבה נפרדת

תהליכון T2 של P1 מבקש פעולת קלט/פלט ארוכת טווח (I/O).

מייד לאחר מכן, מתקבלת מהתקן I/O פסיקת חומרה (Hardware Interrupt), שמעידה על השלמת פעולה קודמת של תהליכון T1 שהיה במצב 'Blocked'.

מהו רצף האירועים הסביר ביותר שיקרה, ומדוע?

1. התהליכון T1 ירוץ מייד, כיוון שהפסיקה קיבלה עדיפות על פני בקשת ה־I/O של T2.
2. התהליך P2 יעבור למצב 'Blocked' עקב בקשת ה־I/O של T2, והמערכת תיכנס למצב של 'Deadlock' בין P1 ל־P2.
3. כל התהליכונים של P1 יעברו למצב 'Blocked', מכיוון שתהליכון אחד שלהם ביקש I/O.
4. התהליכון T2 יעבור למצב 'Blocked'. במקביל, הליבה תטפל בפסיקה, תעיר את תהליכון T1 ותעביר אותו למצב 'Ready', והמתזמן יבחר את התהליך הבא לריצה.

שאלה 25

נתונה מערכת מרובת ליבות עם היררכיית זיכרון מלאה (כולל TLB וזיכרון מטמון L1, L2).

תהליך במצב 'Running' על ליבה מסוימת חווה Page Fault.

הדף הנדרש נמצא בזיכרון הווירטואלי אך אינו טעון ב־RAM, ואינו נחשב 'מלוכלך' (Dirty).

במקביל, ה־TLB של הליבה מכיל נתונים שהפכו ללא תקפים (Invalid) בעקבות שינוי בטבלת הדפים על ידי ליבה אחרת.

מהו רצף האירועים הנכון ביותר שיקרה, ומה ההשלכה המרכזית על ביצועי המערכת?

1. ה־Page Fault יטופל תחילה על ידי טעינת הדף מהדיסק, ובמקביל, הליבה האחרת תעדכן את ה־TLB.
2. ה־TLB מוצא כתובת לא תקינה ומעלה חריגה (Exception), המעבד עוצר את ריצת התהליך, ה־TLB מנוקה והבעיה מתוקנת, ואז המעבד מזהה את ה־Page Fault ומתחיל את הטיפול בו על ידי העברת התהליך למצב 'Blocked'.
3. המעבד מזהה את ה־Page Fault, מעביר את התהליך למצב 'Blocked'. ומערכת ההפעלה מטפלת בטעינת הדף. התקלה ב־TLB אינה משמעותית כיוון שהיא תטופל בהמשך.
4. המערכת תבצע החלפת תהליך (Context Switch) לתהליך אחר בעל עדיפות גבוהה יותר, והבעיה תטופל לאחר מכן.

המשך בעמוד 13

חלק ב': תקשורת נתונים ורשתות מחשבים (50 נקודות)

שימו לב: את כל התשובות יש לסמן אך ורק על גבי דף התשובות שבנספח א'.
ענו על שלוש השאלות 26-28.

שאלה 26 (15 נקודות)

ענו על כל הסעיפים א'-י' (לכל סעיף – 1.5 נק')

לפניכם קוד צד שרת וקוד צד לקוח DNS. התבוננו בו וענו על הסעיפים המופיעים אחריו.

צד שרת:

```
1.   #include <stdio.h>
2.   #include <stdlib.h>
3.   #include <string.h>
4.   #include <unistd.h>
5.   #include <arpa/inet.h>
6.
7.   #define DNS_PORT 53
8.   #define BUF_SIZE 512
9.
10.  void build_dns_response(unsigned char* buffer, int query_len) {
11.
12.    buffer[2] |= 0x80;
13.    buffer[3] |= 0x80;
14.    buffer[7] = 1;
15.    int pos = query_len;
16.
17.    memcpy(&buffer[pos], &buffer[12], query_len - 12);
18.    pos += query_len - 12;
19.
20.    // Type A (host address)
21.    buffer[pos++] = 0x00;
22.    buffer[pos++] = 0x01;
23.
24.    // Class IN
25.    buffer[pos++] = 0x00;
26.    buffer[pos++] = 0x01;
27.
28.    // TTL
```

המשך בעמוד 14

```
29.     buffer[pos++] = 0x00;
30.     buffer[pos++] = 0x00;
31.     buffer[pos++] = 0x00;
32.     buffer[pos++] = 0x3c; // 60 seconds
33.
34.     // Data length
35.     buffer[pos++] = 0x00;
36.     buffer[pos++] = 0x04;
37.     buffer[pos++] = 1;
38.     buffer[pos++] = 2;
39.     buffer[pos++] = 3;
40.     buffer[pos++] = 4;
41.   }
42.
53    int main() {
44.     int sockfd;
45.     struct sockaddr_in server_addr, client_addr;
46.     unsigned char buffer[BUF_SIZE];
47.     socklen_t addr_len = sizeof(client_addr);
48.     sockfd = socket(_______(1)_______);
49.
50.   if (sockfd < 0) {
51.         perror("socket failed");
52.         return 1;
53.     }
54.     int opt = 1;
55.     setsockopt(sockfd, SOL_SOCKET, SO_REUSEADDR, &opt, sizeof(opt));
56.     memset(&server_addr, 0, sizeof(server_addr));
57.     server_addr.sin_family = AF_INET;
58.     _________________(2)_____________________
59.     server_addr.sin_addr.s_addr = INADDR_ANY;
60.     if (bind(sockfd, (struct sockaddr*)&server_addr, sizeof(server_addr)) < 0) {
61.        perror("bind failed (are you root?)");
62.        close(sockfd);
63.        return 1;
64.     }
```

המשך בעמוד 15

```
65.    printf("DNS server started on port %d...\n", DNS_PORT);
66.  while (1) {
67.        int len = recvfrom(sockfd, buffer, BUF_SIZE, 0, (struct sockaddr*)&client_
           addr, &addr_len);
68.        if (len < 0) continue;
69.
70.        printf("Received DNS query (%d bytes)\n", len);
71.
72.        build_dns_response(buffer, len);
73.        sendto(sockfd, buffer, len + 16, 0, (struct sockaddr*)&client_addr, addr_
           len);
74.    }
75.
76.    close(sockfd);
77.    return 0;
78.  }
```

79. **צד לקוח:**

```
80.  #include <stdio.h>
81.  #include <stdlib.h>
82.  #include <string.h>
83.  #include <unistd.h>
84.  #include <arpa/inet.h>
85.
86.  #define DNS_PORT 53
87.  #define BUF_SIZE 512
88.
89.  int encode_domain_name(unsigned char *dns, const char *host) {
90.    int len = 0;
91.    const char *pos = host;
92.    const char *dot;
93.
94.    while ((dot = strchr(pos, '.'))) {
95.    int label_len = dot - pos;
96.     *dns++ = label_len;
97.     memcpy(dns, pos, label_len);
98.     dns += label_len;
```

המשך בעמוד 16

```
99.      pos = dot + 1;
100.     len += label_len + 1;
101.   }
102.
103.   int final_len = strlen(pos);
104.   *dns++ = final_len;
105.   memcpy(dns, pos, final_len);
106.   dns += final_len;
107.   *dns++ = 0;
108.   return dns - (unsigned char *)host;
109. }
110.
111. int main() {
112.   int sockfd;
113.   struct sockaddr_in server_addr;
114.   unsigned char buffer[BUF_SIZE];
115.
116.   const char *dns_server_ip = "127.0.0.1";
117.   const char *domain_name = "example.com";
118.   sockfd = socket(AF_INET, SOCK_DGRAM, 0);
119.   if (sockfd < 0) {
120.      perror("socket failed");
121.      return 1;
122.   }
123.
124.   memset(&server_addr, 0, sizeof(server_addr));
125.   server_addr.sin_family = AF_INET;
126.   server_addr.sin_port = htons(DNS_PORT);
127.   inet_pton(AF_INET, dns_server_ip, &server_addr.sin_addr);
128.   memset(buffer, 0, BUF_SIZE);
129.
130.   buffer[0] = 0x12;
131.   buffer[1] = 0x34;
132.   buffer[2] = 0x01
133.   buffer[4] = 0x00;
134.   buffer[5] = _____(3)_____;
```

המשך בעמוד 17

```
135.
136.   int pos = 12;
137.   pos += encode_domain_name(&buffer[pos], domain_name);
138.   buffer[pos++] = 0x00;
139.   buffer[pos++] = 0x01;
140.
141.   buffer[pos++] = 0x00;
142.   buffer[pos++] = 0x01;
143.
144.   sendto(sockfd, buffer, pos, 0, _____(4)_____, sizeof(server_addr));
145.
146.   socklen_t len = sizeof(server_addr);
147. int recv_len = recvfrom(sockfd, buffer, BUF_SIZE, 0, (struct sockaddr *)&server_
     addr, &len);
148. if (recv_len < 0) {
149.   perror("recvfrom failed");
150.   close(sockfd);
151.   return 1;
152. }
153.
154.   printf("Received DNS response (%d bytes)\n", recv_len);
155.
156.
157. for (int i = 12; i < recv_len - 4; i++) {
158.   if (_____(5)_____) {
159.         int ip_offset = i + 12;
160.         printf("Returned IP: %d.%d.%d.%d\n",
161.                buffer[ip_offset], buffer[ip_offset + 1],
162.                buffer[ip_offset + 2], buffer[ip_offset + 3]);
163.         break;
164.     }
165. }
166.
167. close(sockfd);
168. return 0;
169.}
```

המשך בעמוד 18

ענו על הסעיפים שלפניכם על פי הקודים לעיל:

א. בהגדרת סוקט UDP, חסרים בשורה 48 במקום המסומן ב־(1) שני פרמטרים. מהם הפרמטרים החסרים?

1. AF_INET, SOCK_STREAM
2. AF_INET, SOCK_DGRAM
3. SOCK_DGRAM, AF_INET
4. AF_INET6, SOCK_RAW

ב. איזו פקודה יש להשלים בשורה 58 , במקום המסומן ב־(2) , כדי לקבוע את פורט השרת?

1. server_addr.sin_port = htons(53);
2. server_addr.sin_port = ntohs(53);
3. server_addr.sin_port = 53;
4. server_addr.port = htons(53);

ג. מדוע השאילתה שבשורה 60 משתמשת ב־bind?

1. כדי להתחבר ללקוח
2. כדי לשייך את הסוקט לכתובת ולפורט ברשת
3. כדי להפעיל הצפנה
4. כדי לשלוח תגובה אוטומטית

ד. בשורה 89 מופיעה הפונקצייה:
encode_domain_name (unsigned char *dns, const char *host)
מה מטרת הפונקצייה הזו בצד הלקוח?

1. לפענח את תגובת ה־DNS מהשרת ולהמיר אותה למחרוזת רגילה
2. להמיר כתובת IP של שרת DNS לפורמט בינארי
3. להמיר שם תחום URL לפורמט המתאים למשלוח בבקשת DNS
4. ליצור כתובת אקראית עבור בקשת DNS כדי למנוע התקפות

ה. מה תפקיד הפונקצייה inet_pton שבשורה 127 ?

1. להמיר כתובת IP מטקסט לפורמט בינארי
2. לפרש שמות דומיין
3. לאתחל את מבנה הסוקט
4. לשלוח את הבקשה לשרת

המשך בעמוד 19

ו. מה תפקיד ההשמה buffer[2] = 0 × 01 שבשורה 132 ?

1. לסמן שהתגובה תהיה שלילית
2. להגדיר בקשה מסוג QUERY
3. לקבוע שהתגובה תכלול יותר מתשובה אחת
4. להפעיל את הסוקט במצב TCP

ז. איזה ערך יש להשלים בשורה 134 , במקום המסומן ב־(3) , כדי להגדיר בבקשה "שאלה אחת"?

1. 0x00
2. 0xFF
3. 0x01
4. 0x10

ח. איזה פרמטר חסר בשורה 144 , במקום המסומן ב־(4), בפונקצייה sendto?

1. (struct sockaddr_in*)&client_addr
2. NULL
3. (struct sockaddr*)&server_addr
4. buffer + server_addr

ט. הפונקצייה recvfrom שבשורה 147 מחזירה ערך שלילי. מה משמעות הדבר?

1. הבקשה מתועדת ביומן
2. כתובת השרת נוצרת מחדש
3. התוכנית מדפיסה את הבקשה בכל מקרה
4. הבקשה נדחתה

י. איזה תנאי חסר בשורה 158 , במקום המסומן ב־(5) ?

1. buffer[i] == 0x00 && buffer[i+1] == 0xC0
2. buffer[i] == 'C' && buffer[i+1] == '0'
3. buffer[i] == 0xC0 && buffer[i+1] == 0x0C
4. buffer[i] == 0x3C && buffer[i+1] == 0x00

המשך בעמוד 20

שאלות 27–28 מתייחסות בחלקן לתיאור המקרה שלהלן (מופיע גם בנספח ג'), ולטופולוגיה שבנספח ג'.

אוניברסיטת "UniversNet" היא מוסד אקדמי רב־תחומי בעלת שלוחות בפריסה ארצית ובינלאומית. האוניברסיטה חרתה על דגלה להוביל בתחומי ההוראה, המחקר והחדשנות, תוך שימוש בטכנולוגיות מתקדמות, ויצירת קשרים הדוקים עם התעשייה.

האוניברסיטה מפעילה שישה קמפוסים שונים, שכל אחד מהם מתמחה בתחומי ידע ייחודיים: הנדסה, ניהול עסקי, מחקר יישומי, חינוך, טכנולוגיות מידע ועוד. כמו כן, לאוניברסיטה שלוחות בין־לאומיות אשר משמשות גשר לשיתופי פעולה מחקריים לסטודנטים בארץ ובחו"ל.

האוניברסיטה שמה לה למטרה להנגיש את הידע לסטודנטים בכל העולם ומשקיעה בתשתיות מתקדמות לצורך קישור בין הסניפים, תיאום מערכות מידע, סנכרון בין מרצים וסטודנטים, והנגשת שירותים מקוונים.

תשתיות הרשת מתבססות על ציוד תקשורת מתקדם תוך מתן דגש על אבטחת המידע שימוש בענן, גמישות ושרידות.

קיימים שלושה קמפוסים בישראל ושלושה בחו"ל (נספח ג' מכיל את טופולוגית הרשת של UniversNet)

1. ירושלים (LAN–JER), הנהלה ראשית והפקולטות למדעי החברה ומדעי הרוח.
2. תל אביב (LAN–TLV), הפקולטה למדעי המחשב וטכנולוגיות מידע.
3. חיפה (LAN–Haifa), הפקולטות להנדסת בניין, ניהול עסקי והפקולטה לחינוך.
4. ניו־יורק (LAN–NY), תוכניות בינלאומיות ושיתופי פעולה אקדמאיים.
5. פריז (LAN–PARIS), פקולטת מחקר במדעי הרוח ואומנויות בדגש על שיתוף פעולה עם מוסדות באירופה.
6. הונג קונג (LAN–HK), מרכז טכנולוגי וחדשנות חינוכית. בעיקר ללמידה מרחוק.

הנחות יסוד:

- כל הסניפים בישראל: מקבלים כתובות DHCP מהשרת המוסדי המחובר לנתב בסניף ירושלים.

  בסניפים בחו"ל, מוגדר פרוטוקול DHCP בכל אחד מהנתבים.
- פרוטוקול הניתוב EIGRP מוגדר בסניפים בישראל. בחו"ל מוגדר פרוטוקול OSPF.

המשך בעמוד 21

שאלה 27 (20 נקודות)

ענו על <u>כל</u> הסעיפים א–ט"ו (לכל סעיף – $12\frac{1}{3}$ נק').
סעיפים א'–ד' מתייחסים לטופולוגיה שבנספח ג'.

א. בנתב R–HK הוגדרו ה־VLANs למורים (Teachers) והחיבור לנתב.
אילו פקודות יש להגדיר במתג SW2–HK?

.1

```
SW2-HK(config)# vlan 620
SW2-HK(config-vlan)# exit
SW2-HK(config)# interface FastEthernet0/3
SW2-HK(config-if)# switchport mode access
SW2-HK(config-if)# switchport access vlan 620
SW2-HK(config-if)# exit
SW2-HK(config)# interface FastEthernet0/2
SW2-HK(config-if)# switchport mode access
SW2-HK(config-if)# switchport access vlan 620
SW2-HK(config-if)# exit
SW2-HK(config)# interface FastEthernet0/1
SW2-HK(config-if)# switchport mode trunk
SW2-HK(config-if)# exit
```

.2

```
SW2-HK(config)# vlan 610
SW2-HK(config-vlan)# exit
SW2-HK(config)# interface FastEthernet0/2
SW2-HK(config-if)# switchport mode access
SW2-HK(config-if)# switchport access vlan 610
SW2-HK(config-if)# exit
SW2-HK(config)# interface FastEthernet0/2
SW2-HK(config-if)# switchport mode access
SW2-HK(config-if)# switchport access vlan 610
SW2-HK(config-if)# exit
SW2-HK(config)# interface FastEthernet0/1
SW2-HK(config-if)# switchport mode trunk
SW2-HK(config-if)# exit
```

המשך בעמוד 22

.3

```
SW2-HK(config)# vlan 620
SW2-HK(config-vlan)# exit
SW2-HK(config)# interface FastEthernet0/3
SW2-HK(config-if)# switchport mode trunk
SW2-HK(config-if)# switchport access vlan 610
SW2-HK(config-if)# exit
SW2-HK(config)# interface FastEthernet0/4
SW2-HK(config-if)# switchport mode access
SW2-HK(config-if)# switchport access vlan 620
SW2-HK(config-if)# exit
SW2-HK(config)# interface FastEthernet0/1
SW2-HK(config-if)# switchport mode trunk
SW2-HK(config-if)# exit
```

.4

```
SW2-HK(config)# vlan 620
SW2-HK(config-vlan)# exit
SW2-HK(config)# interface FastEthernet0/3
SW2-HK(config-if)# switchport mode trunk
SW2-HK(config-if)# switchport access vlan 620
SW2-HK(config-if)# exit
SW2-HK(config)# interface FastEthernet0/2
SW2-HK(config-if)# switchport mode trunk
SW2-HK(config-if)# switchport access vlan 620
SW2-HK(config-if)# exit
SW2-HK(config)# interface FastEthernet0/1
SW2-HK(config-if)# switchport mode access
SW2-HK(config-if)# exit
```

המשך בעמוד 23

ב. מדוע האוניברסיטה בחרה להגדיר את כתובות ה־IP באמצעות שרת DHCP ולא באופן ידני?

1. מכיוון ששרת DHCP מרכז את חלוקת הכתובות בידי גוף אחד ומונע לחלוטין התנגשויות של כתובות IP.
2. מכיוון ששרת DHCP מרכז את חלוקת הכתובות בידי גוף אחד וממזער היתכנות להתנגשות כתובות.
3. מכיוון ששרת ה־DHCP קיים רק לגיבוי.
4. מכיוון ששרת DHCP מסתיר ממשתמשים שנמצאים מחוץ לרשת את הכתובות האמיתיות של המחשבים ברשת.

ג. לאיזו רשת שייך המחשב Student1 בקמפוס ניו־יורק NY?

1. 192.168.60.0/28
2. 192.168.60.16/28
3. 192.168.60.32/28
4. 192.168.60.48/28

ד. באוניברסיטה הגדירו את הסיב האופטי שבין ירושלים לתל אביב כ־multimode, ואת הסיב האופטי שבין תל אביב להונג קונג כ־singlemode בטכנולוגיית מטרו. מה ההבדל בין שני סוגי הסיבים?

1. ב־singlemode משתמשים להעברת כמות מידע קטנה, וב־multimode להעברת כמות מידע גדולה.
2. ב־singlemode משתמשים להעברת מידע למרחקים גדולים, וב־multimode משתמשים להעברת מידע למרחקים קטנים.
3. singlemode הוא סיב בטכנולוגיה ישנה, ו־multimode הוא סיב בטכנולוגיה חדשה.
4. כל התשובות נכונות

שימו לב: סעיפים ה–ט"ו <u>אינם</u> מתייחסים לטופולוגיה שבנספח ג'.

ה. בחרו במשפט הנכון ביותר לגבי רשת ad hoc.

1. המחשבים ברשת ad hoc מחוברים זה לזה באמצעות Ethernet.
2. כל מחשב ברשת ad hoc יכול לדבר עם מחשבים אחרים כמספר כרטיסי הרשת שהוא מכיל.
3. כל מחשב ברשת ad hoc מתפקד כראוטר וכך הוא שולח נתונים.
4. המחשבים ברשת ad hoc עובדים בשכבה 1 ו־2 בלבד.

ו. לפניכם ארבעה היגדים המתייחסים לשכבה 5 במודל OSI.
קבעו לגבי כל אחד מהם אם הוא **נכון** או **לא נכון**:

1. מכשיר ווקי־טוקי הוא דוגמה קלאסית לתקשורת Half-Duplex. **נכון / לא נכון**
2. שידור רדיו הוא דוגמה לתקשורת Half-Duplex. **נכון / לא נכון**
3. רשת Ethernet מודרנית פועלת בדרך כלל במצב Full-Duplex. **נכון / לא נכון**
4. מקלדת מחשב מתקשרת עם המחשב במצב Simplex. **נכון / לא נכון**

המשך בעמוד 24

ז. איזו שכבה במודל ה־OSI אחראית על תעבורת נתונים אמינה בין שני התקנים המחוברים ישירות, משתמשת במנגנון CRC (Cyclic Redundancy Check) כדי לגלות שגיאות, ומעבירה את הטיפול ב־frame לשכבות גבוהות יותר במודל?

1. שכבת התעבורה (Transport Layer)
2. שכבת הרשת (Network Larer)
3. שכבת קישור הנתונים (Data Link Layer)
4. השכבה הפיזית (Physical Layer)

ח. איזה פרוטוקול בשכבת התעבורה (Transport Layer) נועד להבטיח אמינות באמצעות חלוקת המידע למקטעים (Segments), אישורי קבלה (ACKs) ושידור חוזר (Retransmission) של נתונים שאבדו?

1. Internet Protocol
2. User Datagram Protocol
3. Transmission Control Protocol
4. Internet Control Message Protocol

ט. מה מטרתו של מנגנון CRC (Cyclic Redundancy Check) בשכבה 2 במודל OSI?

1. לוודא שכל מסגרות הנתונים נשלחות בסדר הנכון
2. לשמר גודל קבוע של מסגרת (frame)
3. לבדוק תקינות עבור כל מסגרת (frame) בנפרד
4. כל התשובות נכונות

י. לפניכם כתובת IPv6 מקוצרת: 2001:db8::123:4567 .
מהי הכתובת המלאה התקינה שלה?

1. 8:0:0:0:0:123:4567db2001:0
2. 8:00:000:000000002001:0:0123:4567db:2001
3. 0123:4567::8db2001:0
4. 2001:0db8:0000:0000:0000:0000:0123:4567

יא. איזה מבין השדות הבאים נמצא ב־Header בכתובת IPv4, אך אינו קיים ב־Header בכתובת IPv6?

1. Source Address
2. Destination Address
3. Checksum
4. Version

המשך בעמוד 25

יב. CSMA/CD פחות רלוונטי ברשתות Ethernet מודרניות המבוססות על switch. מדוע?

1. מכיוון ש־CSMA/CD מיועד רק לרשתות אלחוטיות.
2. מכיוון ש־switch אינו תומך בשידור נתונים.
3. מכיוון ש־switch יוצר מתחם התנגשות נפרד לכל פורט, ומאפשר מצב Full-Duplex.
4. מכיוון ש־switch מיישם פרוטוקול גישה למדיום חדש בשם "Switching Access Protocol".

יג. מהן שתי תתי השכבות שמרכיבות את שכבת קישור הנתונים (Data Link Layer) על פי מודל 802IEEE?

1. Ethernet ו־Wi-Fi
2. LLC ו־MAC
3. IP ו־TCP
4. PHY ו־MAC

יד. מה משמעות המונח "זמן התכנסות" (Convergence Time) בהקשר של פרוטוקולי ניתוב דינמיים?

1. הזמן הדרוש למעבר חבילה מנקודת המוצא (Source) לנקודת היעד (Destination).
2. הזמן הדרוש כדי לעדכן את כל טבלאות הניתוב ברשת לאחר שינוי טופולוגי.
3. הזמן הדרוש לנתב "ללמוד" את הרשת כולה.
4. הזמן הדרוש לנתב כדי להתחבר ל־ISP.

טו. לפניכם ארבעה היגדים המתייחסים לפרוטוקולי ניתוב דינמיים.
קבעו לגבי כל אחד מהם אם הוא **נכון** או **לא נכון**:

1. פרוטוקול ניתוב מסוג Link-State (כמו OSPF) נוטה להתכנס (converge) לאט יותר מפרוטוקול מסוג Distance – Vector (כמו RIP) במקרה של שינוי בטופולוגיה, מכיוון שהוא דורש החלפת מידע נרחבת יותר.
   נכון / לא נכון
2. פרוטוקול ניתוב מסוג Distance–Vector אשר מעדכן רק את הנתבים השכנים, דורש פחות משאבי מחשב וזיכרון מאשר פרוטוקול ניתוב מסוג Link–State הדורש זיכרון לאחסון בסיס הנתונים של כל הנתבים בטופולוגיית הרשת המקומית (Link-State Database).
   נכון / לא נכון
3. כל נתב בפרוטוקול Link–State מפיץ מידע רק על הקישורים המחוברים אליו ישירות (LSA, Link–State Advertisements), בעוד בפרוטוקול Distance–Vector הוא מפיץ את טבלת הניתוב המלאה שלו לשכניו בלבד.
   נכון / לא נכון
4. פרוטוקולי Distance–Vector תוכננו מראש לתמוך במבנה היררכי של חלוקה לאזורים (Areas) קטנים ומנוהלים כדי לצמצם את גודל בסיס הנתונים, לעומת פרוטוקולי Link–State שאינם תומכים במבנה זה.
   נכון / לא נכון

המשך בעמוד 26

שאלה 28 (15 נקודות)

ענו על <u>כל</u> הסעיפים מתוך הסעיפים א–ט (לכל סעיף – $1\frac{2}{3}$ נק')
סעיפים א'–ז' מתייחסים לטופולוגיה שבנספח ג'.

א. לפניכם פלט חלקי מנתב R–Haifa.

```
interface GigabitEthernet1/0
ip address 10.48.0.1  255.240.0.0
duplex auto
speed auto
!
interface GigabitEthernet2/0
ip address 10.0.0.2  255.240.0.0
duplex auto
speed auto
!
interface FastEthernet4/0
ip address 10.16.0.1  255.240.0.0
ip helper-address 200.100.50.2
duplex auto
speed auto
!
interface FastEthernet5/0
ip address 10.32.0.1 255.240.0.0
ip helper-address 200.100.50.2
duplex auto
speed auto
```

תחנות העבודה המתחברות לרשת מקבלות את הכתובת : 169.254....
מה הסיבה לכך? הניחו שכל ההגדרות האחרות בנתב תקינות.

1. לא הוגדרה הפקודה ip helper–address בממשק GigabitEthernet2/0
2. הכתובות המופיעות בפקודה ip helper–address לא נכונות ולכן שרת ה־DHCP לא זמין.
3. לא הוגדר בממשקי הרשת המקומית בנתב חיפה: half–duplex.
4. יש לכבות ולהדליק את הנתב ולנסות שוב.

המשך בעמוד 27

ב. בנתב R–NY הוגדר פרוטוקול DHCP.
לפניכם הפקודות שנכתבו ב־CLI לנתב.

```
NY>ena
NY# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
NY(config)#ip dhcp pool RNY-LAN-POOL
NY(dhcp-config)#network _____(1)_____ _____(2)_____
NY(dhcp-config)#default-router _____(3)_____
NY(dhcp-config)#dns-server 200.100.50.2
NY(config)#exit
NY#
%SYS-5-CONFIG_I: Configured from console by console
NY#_____(4)_____
Destination filename [startup-config]?
Building configuration...
[OK]
```

1. לפניכם שלושה תתי־סעיפים, בהתאמה לערכים החסרים בפקודות לנתב המסומנים במספרים (1)-(3).
בכל אחד מתתי הסעיפים יש לבחור ביטוי **אחד** נכון בלבד מבין האפשרויות i-iv .

(1)	**(2)**	**(3)**
i. 192.168.60.16	i. 192.168.60.16	i. 192.168.60.16
ii. 192.168.60.17	ii. 192.168.60.17	ii. 192.168.60.17
iii. 255.255.255.240	iii. 255.255.255.240	iii. 255.255.255.240
iv. 255.255.248.0	iv. 255.255.248.0	iv. 255.255.248.0

2. איזו פקודה יש לכתוב במקום המסומן במספר (4), כדי לשמור את תוכן הגדרת פקודות ה־DHCP שבנתב R–NY בזיכרון ה־NVRAM, כך שהפקודות יישמרו גם לאחר כיבוי הנתב?

1) copy startup–config running–config
2) copy running–config FTP
3) copy RNY-LAN–POOL startup–config
4) copy running–config startup–config

המשך בעמוד 28

ג. מנהל רשת בקמפוס R–TLV מעוניין לחסום תעבורה מ־LAN-NY מלהגיע לרשתות ה־LAN שמחוברות ישירות לנתב R-TLV, ולאפשר תעבורה תקינה לכל שאר החבילות.
הניחו שחבילות הנתונים מ־LAN–NY לנתב R–TLV עוברות דרך R–Haifa. אילו פקודות צריך מנהל הרשת להגדיר?

.1

```
R-TLV(config)#access-list 10 deny 192.168.60.0  0.0.0.15
R-TLV(config)#access-list 10 permit any
R-TLV(config)#interface gigabitEthernet 1/0
R-TLV(config-if)#ip access-group 10 in
```

.2

```
R-TLV(config)#access-list 10 deny 192.168.60.0  0.0.255.255
R-TLV(config)#access-list 10 permit any
R-TLV(config)#interface gigabitEthernet 1/0
R-TLV(config-if)#ip access-group 10 out
```

.3

```
R-TLV(config)#access-list 10 deny 192.168.60.0  0.0.0.15
R-TLV(config)#interface gigabitEthernet 1/0
R-TLV(config-if)#ip access-group 10 in
```

.4

```
R-TLV(config)#access-list 1 permit any
R-TLV(config)#access-list 1 deny 192.168.60.0 0.0.0.15
R-TLV(config)#interface fastEthernet 4/0
R-TLV(config-if)#ip access-group 1 in
```

המשך בעמוד 29

ד. כאשר מחשב Pari–Teacher2 (IP: 192.168.100.66) מבצע PING למחשב Pari–Teacher1 בסניף LAN–Paris, מתקבל הפלט:

```
C:\>ping 192.168.100.130

Pinging 192.168.100.130 with 32 bytes of data:

Reply from 192.168.100.130: bytes=32 time<1ms TTL=127
Reply from 192.168.100.130: bytes=32 time=1ms TTL=127
Reply from 192.168.100.130: bytes=32 time<1ms TTL=127
Reply from 192.168.100.130: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.100.130:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>
```

אך כאשר מחשב PARI–Teacher2 מבצע PING למחשב PARI–Teacher1 באמצעות הפקודה:

```
c:\>ping PARI-Teacher1
```

מתקבל הפלט:

```
C:\>ping PARI-Teacher1
Ping request could not find host PARI-Teacher1. Please check the name and try again
C:\>
```

מדוע?

1. הפקודה "ping PARI–Teacher1" אינה חוקית. ניתן לכתוב אחר ה־ping רק כתובת IP.
2. השם PARI–Teacher1 לא מוגדר בשרת ה־DNS.
3. השם PARI–Teacher2 לא מוגדר בשרת ה־DNS.
4. למחשב PARI-–Teacher1 בנתב PARIS אין כתובת IP תקינה.

המשך בעמוד 30

ה. לאחרונה התגלו ניסיונות גישה חשודים מכתובת ה־IP של Student1 בקמפוס תל אביב לשרת ה־SQL הקריטי של קמפוס ירושלים. גישה בלתי מורשית לשרת זה עלולה לסכן נתוני מחקר סודיים.

מנהל האבטחה של רשת האוניברסיטאות UniversNet נדרש לאבטח נתונים אקדמיים רגישים השמורים בשרת זה – למנוע גישה לשרת ממשתמש Student1 בנתב R–TLV ולאפשר לכל שאר המשתמשים בכל הקמפוסים גישה חופשית אליו.

בקטע שלפניכם חסרים **חמישה** ביטויים, המסומנים במספרים (1) – (5). אחריהם מופיעים חמישה סעיפים, אחד לכל אחד מהביטויים החסרים. בחרו בכל אחד מהם את הערך הנכון.

```
R-TLV> enable
R-TLV# configure terminal
R-TLV(config)# access-list 105 __(1)__ tcp __(2)__ 192.168.10.130  0.0.0.0 eq __(3)__
R-TLV(config)# access-list 105 __(4)__
R-TLV(config)# interface __(5)__
R-TLV(config-if)# ip access-group 105 out
R-TLV(config-if)# end
```

1.
   i. permit ip any any
   ii. deny
   iii. deny ip any any
   iv. permit any

2.
   i. 172.17.0.34 0.0.255.255
   ii. 172.17.0.66 0.0.255.0
   iii. 0.0.0.0. 172.17.30.33
   iv. 172.17.0.34 0.0.0.0

3.
   i. 1300
   ii. 14
   iii. 1433
   iv. 1024

4.
   i. permit ip any any
   ii. deny
   iii. deny ip any any
   iv. permit any

5.
   i. Fa4/0
   ii. Fa5/0
   iii. Gig0/0
   iv. Gig1/0

המשך בעמוד 31

1. בנתב R–Haifa התרחשה פריצה וכמות עצומה של תעבורה לא חוקית (Flooding) גרמה לכשל בנתב. כתוצאה מכך הכריז הנתב על הממשק Gig2/0 כ"לא פעיל" (down).

1. הניחו שכל הנתבים משתמשים בפרוטוקול ניתוב דינמי. איזה נתב יסיים ראשון את תהליך ההתכנסות (router convergence process) ויפסיק לשלוח חבילות דרך הקישור שכשל?
   i. R–HK
   ii. R–JER
   iii. R–TLV
   iv. R–PARIS

2. הניחו שבכל הנתבים בישראל משתמשים בפרוטוקול EIGRP ובכל הנתבים בחו"ל משתמשים בפרוטוקול ניתוב OSPF (פרוטוקול Link–State). מהו תהליך ההתכנסות שיתחיל בנתב R–Haifa, וכיצד הוא יבטיח שהמידע על התקלה יופץ ביעילות?
   i. R–Haifa יסיר את כל הנתבים מטבלת הניתוב שלו, יאפס את הגדרותיו וימתין לעדכון מ־R–TLV.
   ii. R–Haifa ישלח לנתב R–NY הודעת LSA (Link–State Advertisement) מיוחדת המכריזה על הכשל, וכך יפעל כנקודת הכינוס המרכזית.
   iii. R–Haifa יזהה את הכשל בממשק Gig2/0, יעדכן את מסד הנתונים של הקישורים שלו (LSDB) ויפיץ LSA חדש לכל הנתבים באזור, אשר יריצו מחדש את אלגוריתם SPF (Shortest Path First) כדי לחשב מסלולים חלופיים.
   iv. R–Haifa ישלח ל־R–TLV הודעה ספציפית שמעדכנת על הכשל, וכך יימנעו לולאות ניתוב.

2. בתחילת דרכה של האוניברסיטה, הופעלה רשת קטנה בלבד של 10 מחשבים. הנתב של האוניברסיטה השתמש ב־PAT כדי לאפשר למרצים לגשת לאינטרנט דרך כתובת ציבורית אחת. כמו כן, האוניברסיטה אפשרה למרצים לגשת אל שרת האינטרנט הפנימי מהמחשבים הביתיים שלהם. איזו טכניקה הוגדרה בנתב כדי להשיג זאת?

1. Port Forwarding
2. PAT (Port Address Translation)
3. static NAT
4. Dynamic NAT

המשך בעמוד 32

שימו לב: סעיפים ח'–ט' <u>אינם</u> מתייחסים לטופולוגיה שבנספח ג'.

ח. מי מבין הפרוטוקולים OSPF ו־EIGRP מהיר יותר בטיפול בשינויים מהירים בטופולוגיה של הרשת (Convergence), ומדוע?

1. OSPF, מכיוון שהוא משתמש ב־Hello Packets כדי לזהות שכן שנפל, בעוד EIGRP משתמש בטיימר ארוך יותר.
2. EIGRP, מכיוון שהוא מפיץ עדכוני ניתוב חלקיים ומונע שידור של טבלת ניתוב שלמה.
3. OSPF, מכיוון שהוא מבוסס על מצב קישור (Link–State) ומחזיק מפה מלאה של הרשת, בעוד EIGRP מבוסס על וקטור מרחק (Distance Vector) וצריך לשלוח שאילתות לנתבים שכנים.
4. EIGRP, מכיוון שהאלגוריתם שלו יכול לעבור באופן מיידי למסלול גיבוי מוגדר מראש (Feasible Successor) מבלי לבצע חישוב מחדש של כל הרשת, בעוד OSPF חייב לבצע חישוב מחדש של עץ ה־SPF.

ט. לפניכם פלט חלקי של טבלת ניתוב:

```
D      10.0.0.0/8 [90/3072] via 172.16.30.33, 00:36:27, GigabitEthernet3/0
       172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
D         172.16.0.0/16 is a summary, 00:36:28, Null0
C         172.16.30.32/27 is directly connected, GigabitEthernet3/0
D      172.17.0.0/16 [90/28416] via 172.16.30.33, 00:36:27, GigabitEthernet3/0
       192.168.10.0/24 is variably subnetted, 4 subnets, 2 masks
D         192.168.10.0/24 is a summary, 00:36:28, Null0
C         192.168.10.0/26 is directly connected, GigabitEthernet2/0
C         192.168.10.64/26 is directly connected, FastEthernet5/0
C         192.168.10.128/26 is directly connected, FastEthernet4/0
       192.168.20.0/24 is variably subnetted, 2 subnets, 2 masks
D         192.168.20.0/24 is a summary, 00:36:28, Null0
C         192.168.20.0/26 is directly connected, GigabitEthernetl/0
       192.168.60.0/24 is variably subnetted, 2 subnets, 2 masks
D         192.168.60.0/24 is a summary, 00:36:28, Null0
C         192.168.60.0/28 is directly connected, GigabitEthernet0/0
```

1. מהו שם הנתב שפלט זה מייצג את טבלת הניתוב שלו?
   כתבו את התשובה במקום המיועד לכך בנספח א'.

2. איזו פקודה הפיקה את הפלט?
   i. show ip interface brief
   ii. show ip route
   iii. show ip protocols
   iv. show interfaces

המשך בעמוד 33

3. בעמודה השמאלית ביותר בפלט מופיעות האותיות D ו־C.
מה מציינת האות D בעמודה זו, ואיזו רשת מתקשרת דרך ממשק GigabitEthernet3/0?

i. D: רשת שנלמדה באופן דינמי מ־DHCP; הרשת: 172.17.0.0/16 .

ii. D: רשת שהוגדרה ידנית; הרשת: 10.0.0.0/8 .

iii. D: רשת שנלמדה באופן דינמי בפרוטוקול EIGRP; הרשת: 172.17.0.0/16 .

iv. D: רשת שהוגדרה כברירת מחדל (Default Route); הרשת: 10.0.0.0/8 .

4. מהי כתובת ה־IP של ה־Next Hop (הדילוג הבא) שאליו יש לשלוח חבילות כדי להגיע לרשת 172.17.0.0/16 שבטבלת הניתוב?

i. 172.16.30.34

ii. 192.168.10.1

iii. 172.16.30.33

iv. 192.168.20.1

5. על פי הפלט, לאילו רשתות־משנה (SubNet) מחובר הנתב הנוכחי דרך הממשקים FastEthernet4/0 ו־FastEthernet5/0?

i. 192.168.10.64/25 , 192.168.10.128/26

ii. 192.168.10.64/26 , 192.168.10.0/26

iii. 192.168.10.0/26 , 192.168.10.128/26

iv. 192.168.10.65/26 , 192.168.10.1/26

בהצלחה!

זכות היוצרים שמורה למדינת ישראל.
אין להעתיק או לפרסם אלא ברשות משרד החינוך.

נספח א': דף תשובות (2 עמודים)

לשאלון 714913, אביב תשפ"ו

מקום למדבקת נבחן

בכל סעיף הקיפו בעיגול את הספרה המייצגת את התשובה הנכונה.
הדביקו את מדבקת הנבחן שלכם במקום המיועד לכך, והדקו את הדף הזה אל מחברת הבחינה.

חלק א' – עקרונות מערכות הפעלה (50 נקודות)

פרק ראשון (30 נקודות)

שאלה 1

סעיף א 1 2 3 4

סעיף ב 1 2 3 4

שאלה 2

1 2 3 4

שאלה 3

1 2 3 4

שאלה 4

1 2 3 4

שאלה 5

1 2 3 4

שאלה 6

1 2 3 4

שאלה 7

1 2 3 4

שאלה 8

1 2 3 4

שאלה 9

1 2 3 4

שאלה 10

1 2 3 4

שאלה 11

1 2 3 4

שאלה 12

1 2 3 4

שאלה 13

1 2 3 4

שאלה 14

1 2 3 4

שאלה 15

1 2 3 4

פרק שני (20 נקודות)

שאלה 16

1 2 3 4

שאלה 17

1 2 3 4

שאלה 18

1 2 3 4

שאלה 19

1 2 3 4

שאלה 20

1 2 3 4

שאלה 21

1 2 3 4

שאלה 22

1 2 3 4

שאלה 23

1 2 3 4

שאלה 24

1 2 3 4

שאלה 25

1 2 3 4

המשך בעמוד 2

חלק ב' – רשתות מחשבים ותקשורת נתונים (50 נקודות)

שאלה 26 (15 נקודות)

סעיף א	**1**	**2**	**3**	**4**
סעיף ב	**1**	**2**	**3**	**4**
סעיף ג	**1**	**2**	**3**	**4**
סעיף ד	**1**	**2**	**3**	**4**
סעיף ה	**1**	**2**	**3**	**4**
סעיף ו	**1**	**2**	**3**	**4**
סעיף ז	**1**	**2**	**3**	**4**
סעיף ח	**1**	**2**	**3**	**4**
סעיף ט	**1**	**2**	**3**	**4**
סעיף י	**1**	**2**	**3**	**4**

שאלה 27 (20 נקודות)

סעיף א	**1**	**2**	**3**	**4**
סעיף ב	**1**	**2**	**3**	**4**
סעיף ג	**1**	**2**	**3**	**4**
סעיף ד	**1**	**2**	**3**	**4**
סעיף ה	**1**	**2**	**3**	**4**

סעיף ו

1. נכון / לא נכון
2. נכון / לא נכון
3. נכון / לא נכון
4. נכון / לא נכון

סעיף ז	**1**	**2**	**3**	**4**
סעיף ח	**1**	**2**	**3**	**4**
סעיף ט	**1**	**2**	**3**	**4**
סעיף י	**1**	**2**	**3**	**4**
סעיף י"א	**1**	**2**	**3**	**4**
סעיף י"ב	**1**	**2**	**3**	**4**
סעיף י"ג	**1**	**2**	**3**	**4**
סעיף י"ד	**1**	**2**	**3**	**4**

סעיף ט"ו

1. נכון / לא נכון
2. נכון / לא נכון
3. נכון / לא נכון
4. נכון / לא נכון

שאלה 28 (15 נקודות)

סעיף א	**1**	**2**	**3**	**4**

סעיף ב

1.

(1) i ii iii iv

(2) i ii iii iv

(3) i ii iii iv

2. 1) 2) 3) 4)

סעיף ג	**1**	**2**	**3**	**4**
סעיף ד	**1**	**2**	**3**	**4**

סעיף ה

1. i ii iii iv
2. i ii iii iv
3. i ii iii iv
4. i ii iii iv
5. i ii iii iv

סעיף ו

1. i ii iii iv
2. i ii iii iv

סעיף ז	**1**	**2**	**3**	**4**
סעיף ח	**1**	**2**	**3**	**4**

סעיף ט

1. _____
2. i ii iii iv
3. i ii iii iv
4. i ii iii iv
5. i ii iii iv

נספח ב':
נספח פקודות לחלק א' – עקרונות מערכות הפעלה (14 עמודים)
לשאלון 714913, אביב תשפ"ו

GREP(1) **User Commands**

NAME

grep - print lines that match patterns

SYNOPSIS

grep [OPTION...] PATTERNS [FILE...]

grep [OPTION...] **-e** PATTERNS ... [FILE...]

grep [OPTION...] **-f** PATTERN_FILE ... [FILE...]

DESCRIPTION

grep searches for PATTERNS in each FILE. PATTERNS is one or more patterns separated by newline characters, and **grep** prints each line that matches a pattern. Typically PATTERNS should be quoted when **grep** is used in a shell command.

A FILE of "**-**" stands for standard input. If no FILE is given, recursive searches examine the working directory, and nonrecursive searches read standard input.

OPTIONS

Generic Program Information

--help Output a usage message and exit.

-V, --version

Output the version number of grep and exit.

Pattern Syntax

-E, --extended-regexp

Interpret PATTERNS as extended regular expressions (EREs, see below).

המשך בעמוד 2

-F, --fixed-strings

Interpret PATTERNS as fixed strings, not regular expressions.

-G, --basic-regexp

Interpret PATTERNS as basic regular expressions (BREs, see below). This is the default.

-P, --perl-regexp

Interpret PATTERNS as Perl-compatible regular expressions (PCREs).
This option is experimental when combined with the **-z** (**--null-data**) option, and **grep -P** may warn of unimplemented features.

Matching Control

-e PATTERNS, **--regexp=**PATTERNS

Use PATTERNS as the patterns. If this option is used multiple times or is combined with the **-f** (**--file**) option, search for all patterns given. This option can be used to protect a pattern beginning with "**-**".

-f FILE, **--file**=FILE

Obtain patterns from FILE, one per line. If this option is used multiple times or is combined with the **-e** (**--regexp**) option, search for all patterns given. The empty file contains zero patterns, and therefore matches nothing.
If FILE is **-** , read patterns from standard input.

-i, --ignore-case

Ignore case distinctions in patterns and input data, so that characters that differ only in case match each other.

--no-ignore-case

Do not ignore case distinctions in patterns and input data. This is the default. This option is useful for passing to shell scripts that already use -i, to cancel its effects because the two options override each other.

-v, --invert-match

Invert the sense of matching, to select non-matching lines.

-w, --word-regexp

Select only those lines containing matches that form whole words. The test is that the matching substring must either be at the beginning of the line, or preceded by a non-word constituent character. Similarly, it must be either at the end of the line or followed by a non-word constituent character. Word-constituent characters are letters, digits, and the underscore. This option has no effect if **-x** is also specified.

המשך בעמוד 3

-x, --line-regexp

Select only those matches that exactly match the whole line. For a regular expression pattern, this is like parenthesizing the pattern and then surrounding it with **^** and **$**.

REGULAR EXPRESSIONS

A regular expression is a pattern that describes a set of strings. Regular expressions are constructed analogously to arithmetic expressions, by using various operators to combine smaller expressions.

grep understands three different versions of regular expression syntax: "basic"(BRE), "extended" (ERE) and "perl" (PCRE). In GNU **grep**, basic and extended regular expressions are merely different notations for the same pattern-matching functionality. In other implementations, basic regular expressions are ordinarily less powerful than extended, though occasionally it is the other way around. The following description applies to extended regular expressions; differences for basic regular expressions are summarized afterwards. Perl-compatible regular expressions have different functionality, and are documented in pcre2syntax(3) and pcre2pattern(3), but work only if PCRE support is enabled.

The fundamental building blocks are the regular expressions that match a single character. Most characters, including all letters and digits, are regular expressions that match themselves. Any meta-character with special meaning may be quoted by preceding it with a backslash.

The period. matches any single character. It is unspecified whether it matches an encoding error.

Character Classes and Bracket Expressions

A bracket expression is a list of characters enclosed by **[** and **]**. It matches any single character in that list. If the first character of the list is the caret **^** then it matches any character not in the list; it is unspecified whether it matches an encoding error. For example, the regular expression **[0123456789]** matches any single digit.

Within a bracket expression, a range expression consists of two characters separated by a hyphen. In the default C locale, it matches any single character that appears between the two characters in ASCII order, inclusive. For example, **[a-d]** is equivalent to **[abcd]**. In other locales the behavior is unspecified: **[a-d]** might be equivalent to **[abcd]** or **[aBbCcDd]** or some other bracket expression, or it might fail to match any character, or the set of characters that it matches might be erratic, or it might be invalid. To obtain the traditional interpretation of bracket expressions, you can use the C locale by setting the **LC_ALL** environment variable to the value **C**.

Finally, certain named classes of characters are predefined within bracket expressions, as follows. Their names are self explanatory, and they are **[:alnum:]**, **[:alpha:]**, **[:blank:]**, **[:cntrl:]**, **[:digit:]**, **[:graph:]**, **[:lower:]**, **[:print:]**, **[:punct:]**, **[:space:]**, **[:upper:]**, and **[:xdigit:]**. For example, **[[:alnum:]]** means the character class of numbers and letters in the current locale. In the C locale and ASCII character set encoding, this is the same as **[0-9A-Za-z]**. (Note that the brackets in these class names are part of the symbolic names, and must be included in addition to the brackets delimiting the bracket expression.) Most meta-characters lose their special meaning inside bracket expressions. To include a literal **]** place it first in the list. Similarly, to include a literal **^** place it anywhere but first. Finally, to include a literal **-** place it last.

Anchoring

The caret **^** and the dollar sign **$** are meta-characters that respectively match the empty string at the beginning and end of a line.

המשך בעמוד 4

The Backslash Character and Special Expressions

The symbols **\<** and **\>** respectively match the empty string at the beginning and end of a word. The symbol **\b** matches the empty string at the edge of a word, and **\B** matches the empty string provided it's not at the edge of a word. The symbol **\w** is a synonym for **[_[:alnum:]]** and **\W** is a synonym for **[^_[:alnum:]]**.

Repetition

A regular expression may be followed by one of several repetition operators:

? The preceding item is optional and matched at most once.

* The preceding item will be matched zero or more times.

+ The preceding item will be matched one or more times.

{n} The preceding item is matched exactly n times.

{n,} The preceding item is matched n or more times.

{m,} The preceding item is matched at most m times. This is a GNU extension.

{n,m} The preceding item is matched at least n times, but not more than m times.

Concatenation

Two regular expressions may be concatenated; the resulting regular expression matches any string formed by concatenating two substrings that respectively match the concatenated expressions.

Alternation

Two regular expressions may be joined by the infix operator **|**; the resulting regular expression matches any string matching either alternate expression.

Precedence

Repetition takes precedence over concatenation, which in turn takes precedence over alternation. A whole expression may be enclosed in parentheses to override these precedence rules and form a subexpression.

Back-references and Subexpressions

The back-reference \n, where n is a single digit, matches the substring previously matched by the nth parenthesized subexpression of the regular expression.

Basic vs Extended Regular Expressions

In basic regular expressions the meta-characters **?**, **+**, **{**, **|**, (, and) lose their special meaning; instead use the backslashed versions **\?**, **\+**, \{, \|, \(, and \).

המשך בעמוד 5

umask(2) **System Calls Manual**

NAME

umask - set file mode creation mask

LIBRARY

Standard C library (libc, -lc)

SYNOPSIS

#include <sys/stat.h>

mode_t umask(mode_t mask**);**

DESCRIPTION

umask() sets the calling process's file mode creation mask (umask) to mask & 0777 (i.e., only the file permission bits of mask are used), and returns the previous value of the mask.

The umask is used by open(2), mkdir(2), and other system calls that create files to modify the permissions placed on newly created files or directories. Specifically, permissions in the umask are turned off from the mode argument to open(2) and mkdir(2).

Alternatively, if the parent directory has a default ACL (see acl(5)), the umask is ignored, the default ACL is inherited, the permission bits are set based on the inherited ACL, and permission bits absent in the mode argument are turned off. For example, the following default ACL is equivalent to a umask of 022:

u::rwx,g::r-x,o::r-x

Combining the effect of this default ACL with a mode argument of 0666 (rw-rw-rw-), the resulting file permissions would be 0644 (rw-r--r--).

The constants that should be used to specify mask are described in inode(7).

The typical default value for the process umask is **S_IWGRP | S_IWOTH** (octal 022). In the usual case where the mode argument to open(2) is specified as:

S_IRUSR | S_IWUSR | S_IRGRP | S_IWGRP | S_IROTH | S_IWOTH

(octal 0666) when creating a new file, the permissions on the resulting file will be:

המשך בעמוד 6

S_IRUSR | S_IWUSR | S_IRGRP | S_IROTH

(because 0666 & ~022 = 0644; i.e. rw-r--r--).

……

SORT(1) User Commands

NAME

sort - sort lines of text files

SYNOPSIS

sort [OPTION]... [FILE]...

sort [OPTION]... --files0-from=F

DESCRIPTION

Write sorted concatenation of all FILE(s) to standard output.

With no FILE, or when FILE is -, read standard input.

Mandatory arguments to long options are mandatory for short options too. Ordering options:

-b, --ignore-leading-blanks

ignore leading blanks

-d, --dictionary-order

consider only blanks and alphanumeric characters

-f, --ignore-case

fold lower case to upper case characters

-g, --general-numeric-sort

compare according to general numerical value

-i, --ignore-nonprinting

consider only printable characters

המשך בעמוד 7

-M, --month-sort

compare (unknown) < 'JAN' < ... < 'DEC'

-h, --human-numeric-sort

compare human readable numbers (e.g., 2K 1G)

-n, --numeric-sort

compare according to string numerical value; see manual for which strings are supported

-R, --random-sort

shuffle, but group identical keys. See shuf(1)

--random-source=FILE

get random bytes from FILE

-r, --reverse

reverse the result of comparisons

--sort=WORD

sort according to WORD: general-numeric **-g**, human-numeric **-h**, month **-M**, numeric **-n**, random **-R**, version **-V**

-V, --version-sort

natural sort of (version) numbers within text

Other options:

--batch-size=NMERGE

merge at most NMERGE inputs at once; for more use temp files

-c, --check, --check=diagnose-first

check for sorted input; do not sort

-C, --check=quiet, **--check**=silent

like -c, but do not report first bad line

המשך בעמוד 8

--compress-program=PROG

compress temporaries with PROG; decompress them with PROG **-d**

--debug

annotate the part of the line used to sort, and warn about questionable usage to stderr

--files0-from=F

read input from the files specified by NUL-terminated names in file F; If F is - then read names from standard input

-k, --key=KEYDEF

sort via a key; KEYDEF gives location and type

-m, --merge

merge already sorted files; do not sort

-o, --output=FILE

write result to FILE instead of standard output

-s, --stable

stabilize sort by disabling last-resort comparison

-S, --buffer-size=SIZE

use SIZE for main memory buffer

-t, --field-separator=SEP

use SEP instead of non-blank to blank transition

-T, --temporary-directory=DIR

use DIR for temporaries, not $TMPDIR or /tmp; multiple options specify multiple directories

--parallel=N

change the number of sorts run concurrently to N

המשך בעמוד 9

-u, --unique

with **-c**, check for strict ordering; without **-c**, output only the first of an equal run

-z, --zero-terminated

line delimiter is NUL, not newline

--help

display this help and exit

--version

output version information and exit

KEYDEF is F[.C][OPTS][,F[.C][OPTS]] for start and s position, where F is a field number and C a character position in the field; both are origin 1, and the s position defaults to the line's end. If neither **-t** nor **-b** is in effect, characters in a field are counted from the beginning of the preceding whitespace. OPTS is one or more single-letter ordering options [bdfgiMhnRrV], which override global ordering options for that key. If no key is given, use the entire line as the key. Use **--debug** to diagnose incorrect key usage.

SIZE may be followed by the following multiplicative suffixes: % 1% of memory, b 1, K 1024 (default), and so on for M, G, T, P, E, Z, Y, R, Q.

*** WARNING *** The locale specified by the environment affects sort order. Set LC_ALL=C to get the traditional sort order that uses native byte values.

……..

TAIL(1) User Commands

NAME

tail - output the last part of files

SYNOPSIS

tail [OPTION]... [FILE]...

DESCRIPTION

Print the last 10 lines of each FILE to standard output. With more than one FILE, precede each with a header giving the file name.

המשך בעמוד 10

With no FILE, or when FILE is -, read standard input.

Mandatory arguments to long options are mandatory for short options too.

-c, --bytes=[+]NUM

output the last NUM bytes; or use -c +NUM to output starting with byte NUM of each file

-f, --follow[={name|descriptor}]

output appended data as the file grows;

an absent option argument means 'descriptor'

-F same as **--follow**=name --retry

-n, --lines=[+]NUM

output the last NUM lines, instead of the last 10; or use **-n** +NUM to skip NUM-1 lines at the start **--max-unchanged-stats**=N with **--follow**=name, reopen a FILE which has not changed size after N (default 5) iterations to see if it has been unlinked or renamed (this is the usual case of rotated log files); with inotify, this option is rarely useful

--pid=PID

with **-f**, terminate after process ID, PID dies; can be repeated to watch multiple processes

-q, --quiet, --silent

never output headers giving file names

--retry

keep trying to open a file if it is inaccessible

-s, --sleep-interval=N

with **-f**, sleep for approximately N seconds (default 1.0) between iterations; with inotify and **--pid**=P, check process P at least once every N seconds

המשך בעמוד 11

-v, --verbose

always output headers giving file names

-z, --zero-terminated

line delimiter is NUL, not newline

--help display this help and exit

--version

output version information and exit

NUM may have a multiplier suffix: b 512, kB 1000, K 1024, MB 1000*1000, M 1024*1024, GB 1000*1000*1000, G 1024*1024*1024, and so on for T, P, E, Z, Y, R, Q. Binary prefixes can be used, too: KiB=K, MiB=M, and so on.

With **--follow** (**-f**), tail defaults to following the file descriptor, which means that even if a tail'ed file is renamed, tail will continue to track its end. This default behavior is not desirable when you really want to track the actual name of the file, not the file descriptor (e.g., log rotation). Use --follow=name in that case. That causes tail to track the named file in a way that accommodates renaming, removal and creation.

……

FALLOCATE(1) **User Commands**

NAME

fallocate - preallocate or deallocate space to a file

SYNOPSIS

fallocate [**-c**|**-p**|**-z**] [**-o** offset] **-l** length [**-n**] filename

fallocate -d [**-o** offset] [**-l** length] filename

fallocate -x [**-o** offset] **-l** length filename

המשך בעמוד 12

DESCRIPTION

fallocate is used to manipulate the allocated disk space for a file, either to deallocate or preallocate it. For filesystems which support the fallocate(2) system call, preallocation is done quickly by allocating blocks and marking them as uninitialized, requiring no IO to the data blocks. This is much faster than creating a file by filling it with zeroes.

The exit status returned by **fallocate** is 0 on success and 1 on failure.

OPTIONS

The length and offset arguments may be followed by the multiplicative suffixes KiB (=1024), MiB (=1024*1024), and so on for GiB, TiB, PiB, EiB, ZiB, and YiB (the "iB" is optional, e.g., "K" has the same meaning as "KiB") or the suffixes KB (=1000), MB (=1000*1000), and so on for GB, TB, PB, EB, ZB, and YB.

The options **--collapse-range**, **--dig-holes**, **--punch-hole**, and **--zero-range** are mutually exclusive.

-c, --collapse-range

Removes a byte range from a file, without leaving a hole. The byte range to be collapsed starts at offset and continues for length bytes. At the completion of the operation, the contents of the file starting at the location offset+length will be appended at the location offset, and the file will be length bytes smaller. The option **--keep-size** may not be specified for the collapse-range operation.

Available since Linux 3.15 for ext4 (only for extent-based files) and XFS. A filesystem may place limitations on the granularity of the operation, in order to ensure efficient implementation. Typically, offset and length must be a multiple of the filesystem logical block size, which varies according to the filesystem type and configuration. If a filesystem has such a requirement, the operation will fail with the error **EINVAL** if this requirement is violated.

-d, --dig-holes

Detect and dig holes. This makes the file sparse in-place, without using extra disk space. The minimum size of the hole depends on filesystem I/O block size (usually 4096 bytes). Also, when using this option, **--keep-size** is implied. If no range is specified by **--offset** and **--length**, then the entire file is analyzed for holes.

You can think of this option as doing a "**cp --sparse**" and then renaming the destination file to the original, without the need for extra disk space.

See **--punch-hole** for a list of supported filesystems.

-i, --insert-range

Insert a hole of length bytes from offset, shifting existing data.

-l, --length length

Specifies the length of the range, in bytes.

המשך בעמוד 13

-n, --keep-size

Do not modify the apparent length of the file. This may effectively allocate blocks past EOF, which can be removed with a truncate.

-o, --offset offset

Specifies the beginning offset of the range, in bytes.

-p, --punch-hole

Deallocates space (i.e., creates a hole) in the byte range starting at offset and continuing for length bytes. Within the specified range, partial filesystem blocks are zeroed, and whole filesystem blocks are removed from the file. After a successful call, subsequent reads from this range will return zeroes. This option may not be specified at the same time as the **--zero-range** option. Also, when using this option, **--keep-size** is implied.

Supported for XFS (since Linux 2.6.38), ext4 (since Linux 3.0), Btrfs (since Linux 3.7), tmpfs (since Linux 3.5) and gfs2 (since Linux 4.16).

-v, --verbose

Enable verbose mode.

-x, --posix

Enable POSIX operation mode. In that mode allocation operation always completes, but it may take longer time when fast allocation is not supported by the underlying filesystem.

-z, --zero-range

Zeroes space in the byte range starting at offset and continuing for length bytes. Within the specified range, blocks are preallocated for the regions that span the holes in the file. After a successful call, subsequent reads from this range will return zeroes.

Zeroing is done within the filesystem preferably by converting the range into unwritten extents. This approach means that the specified range will not be physically zeroed out on the device (except for partial blocks at the either end of the range), and I/O is (otherwise) required only to update metadata.

Option **--keep-size** can be specified to prevent file length modification.

Available since Linux 3.14 for ext4 (only for extent-based files) and XFS.

-h, --help

Display help text and exit.

המשך בעמוד 14

-V, --version

Print version and exit.

……

Information Stored in Page Table Entry

- Frame Number
- V - valid/invalid
- R - read, write (Protection bit)
- M - Modified Bit

נספח ג׳: טופולוגיה לחלק ב׳ (2 עמודים)

לשאלון 714913, אביב תשפ״ו

תיאור הטופולוגיה

אוניברסיטת "UniversNet" היא מוסד אקדמי רב־תחומי בעלת שלוחות בפריסה ארצית ובינלאומית. האוניברסיטה חרתה על דגלה להוביל בתחומי ההוראה, המחקר והחדשנות, תוך שימוש בטכנולוגיות מתקדמות, ויצירת קשרים הדוקים עם התעשייה.

האוניברסיטה מפעילה שישה קמפוסים שונים, שכל אחד מהם מתמחה בתחומי ידע ייחודיים: הנדסה, ניהול עסקי, מחקר יישומי, חינוך, טכנולוגיות מידע ועוד. כמו כן, לאוניברסיטה שלוחות בין־לאומיות אשר משמשות גשר לשיתופי פעולה מחקריים לסטודנטים בארץ ובחו״ל.

האוניברסיטה שמה לה למטרה להנגיש את הידע לסטודנטים בכל העולם ומשקיעה בתשתיות מתקדמות לצורך קישור בין הסניפים, תיאום מערכות מידע, סנכרון בין מרצים וסטודנטים, והנגשת שירותים מקוונים.

תשתיות הרשת מתבססות על ציוד תקשורת מתקדם תוך מתן דגש על אבטחת המידע שימוש בענן, גמישות ושרידות.

קיימים שלושה קמפוסים בישראל ושלושה בחו״ל.

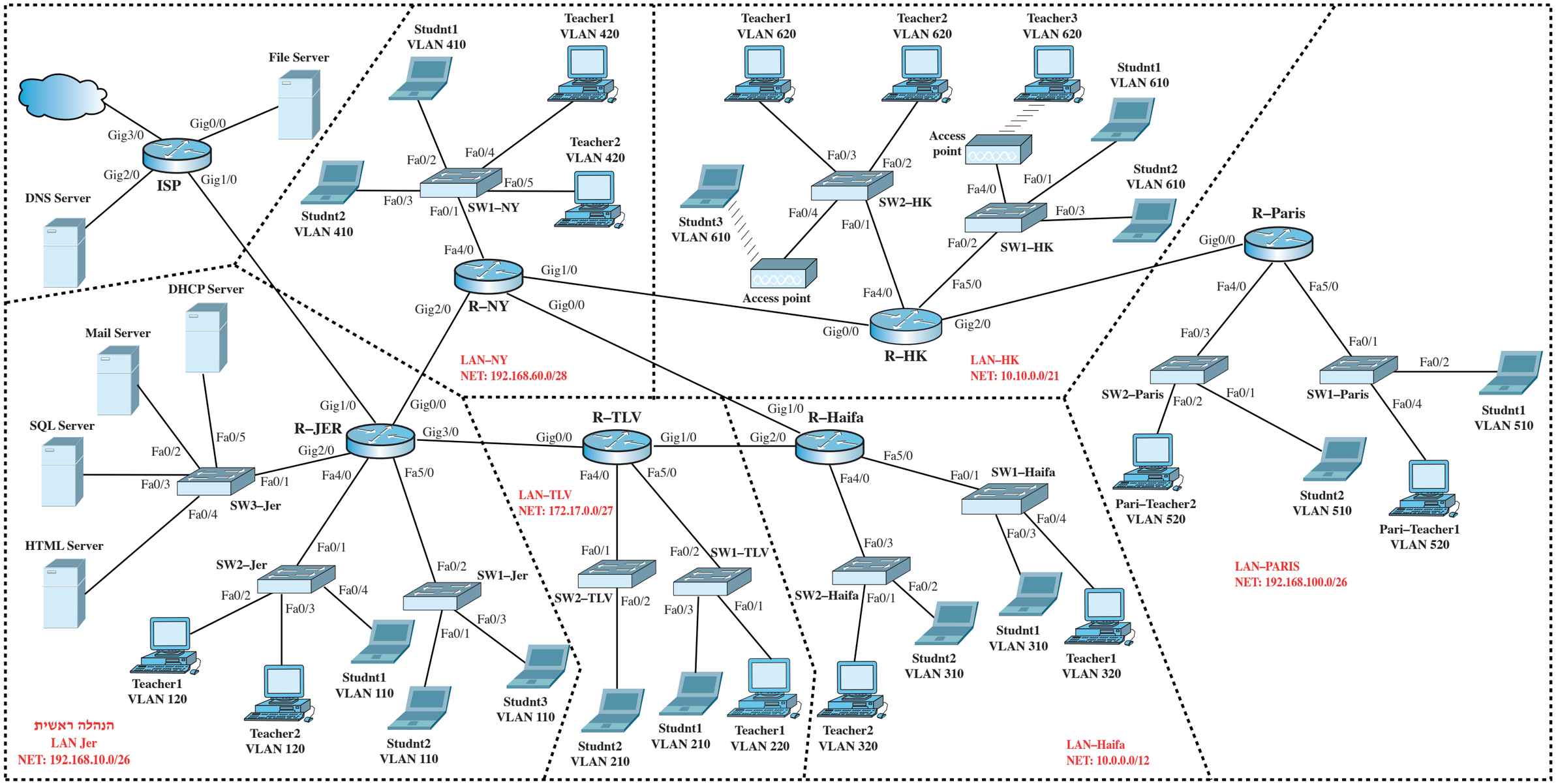
1. ירושלים (LAN–JER), הנהלה ראשית והפקולטות למדעי החברה ומדעי הרוח.
2. תל אביב (LAN–TLV), הפקולטה למדעי המחשב וטכנולוגיות מידע.
3. חיפה (LAN–Haifa), הפקולטות להנדסת בניין, ניהול עסקי והפקולטה לחינוך.
4. ניו־יורק (LAN–NY), תוכניות בינלאומיות ושיתופי פעולה אקדמאיים.
5. פריז (LAN–PARIS), פקולטת מחקר במדעי הרוח ואומנויות בדגש על שיתוף פעולה עם מוסדות באירופה.
6. הונג קונג (LAN–HK), מרכז טכנולוגי וחדשנות חינוכית. בעיקר ללמידה מרחוק.

הנחות יסוד:

- כל הסניפים בישראל: מקבלים כתובות DHCP מהשרת המוסדי המחובר לנתב בסניף ירושלים.
  בסניפים בחו״ל, מוגדר פרוטוקול DHCP בכל אחד מהנתבים.
- פרוטוקול הניתוב EIGRP מוגדר בסניפים בישראל. בחו״ל מוגדר פרוטוקול OSPF.

המשך בעמוד 2

הטופולוגיה:



פירוט כתובות הממשקים בנתבים:

R–Jer	R–TLV	R–Haifa	R–NY	R–HK	R–Paris	ISP
Gig 0/0 – 192.168.60.2/28	Gig 0/0 – 172.16.30.33/27	Gig 1/0 – 10.48.0.1/12	Gig 0/0 – 10.48.0.2/12	Gig 0/0 – 192.168.60.34/28	Gig 0/0 – 10.10.16.2/21	Gig 0/0 – 200.100.50.65 /26
Gig 1/0 – 192.168.20.1 /26	Gig 1/0 – 10.0.0.1/12	Gig 2/0 – 10.0.0.2/12	Gig 1/0 – 192.168.60.33/28	Gig 2/0 – 10.10.16.1/21	Fa4/0 – 192.168.100.65 /26	Gig 1/0 – 192.168.20.2 /26
Gig 2/0 – 192.168.10.1/26	Fa4/0 – 172.17.0.65/27	Fa4/0 – 10.16.0.1/12	Gig 2/0 – 192.168.60.1/28	Fa4/0 – 10.10.24.1/21	Fa5/0 – 192.168.100.129/26	Gig 2/0 – 200.100.50.1/26
Gig 3/0 – 172.16.30.34/27	Fa5/0 - 172.17.0.33/27	Fa5/0 – 10.32.0.1/12	Fa4/0 – 192.168.60.17/28	Fa5/0 – 10.10.32.1/21		Gig 3/0 – 200.100.50.129/26
Fa4/0 – 192.168.10.129/26						
Fa5/0 – 192.168.10.65/26						

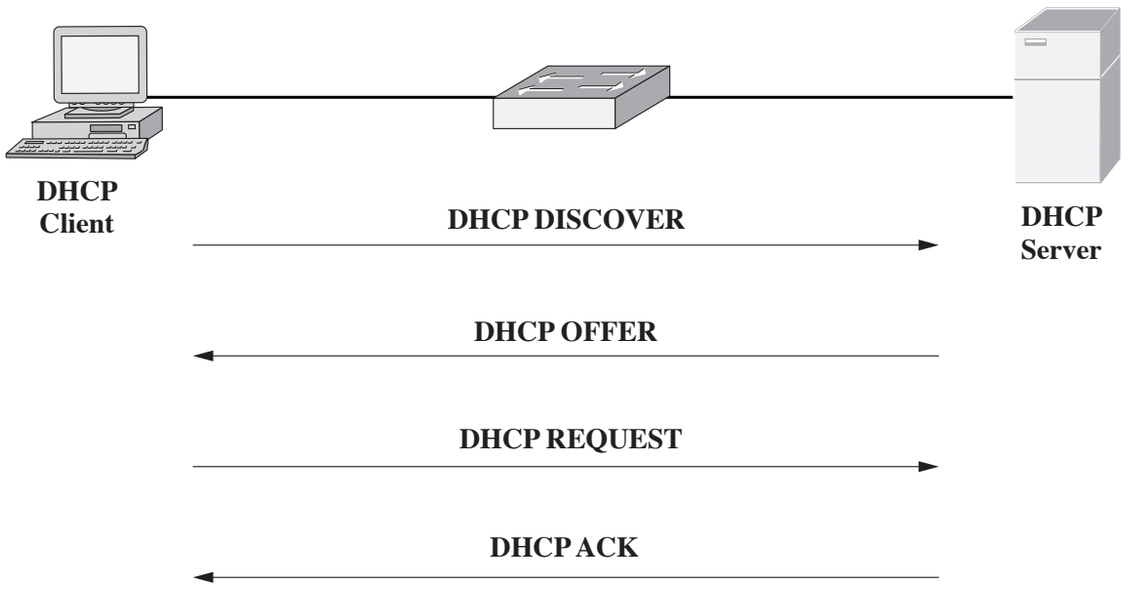
נספח ד':

נספח פקודות לחלק ב' – תקשורת נתונים ורשתות מחשבים (6 עמודים)

לשאלון 714913, אביב תשפ"ו

DHCP

DHCP הוא פרוטוקול תקשורת המאפשר למכשירים ברשת לקבל כתובות IP באופן דינמי ואוטומטי. במקום להגדיר ידנית כתובת IP לכל מכשיר, שרת DHCP מקצה כתובות IP זמניות מתוך מאגר כתובות זמינות.



DHCP Discover

כאשר מכשיר חדש מתחבר לרשת, הוא משדר הודעת גילוי DHCP כדי למצוא שרת DHCP. ההודעה משודרת בשידור רחב (broadcast) לכל המכשירים ברשת.

ההודעה כוללת את הפרטים הבאים:

- כתובת ה־**MAC** של המכשיר: כתובת ייחודית המזהה את המכשיר ברשת.
- שם המכשיר (אופציונלי): שם המכשיר כפי שהוא מוגדר במערכת ההפעלה.
- בקשה לכתובת **IP**: בקשה לקבל כתובת IP משרת DHCP.

שרת DHCP שמקבל את ההודעה בודק את הפרטים ומציע למכשיר כתובת IP זמינה.

DHCP Offer

שרת DHCP שמקבל את הודעת הגילוי מהמכשיר החדש, בודק את הפרטים ומציע לו כתובת IP זמינה. ההצעה כוללת את הפרטים הבאים:

- כתובת ה־**IP** המוצעת: כתובת IP פנויה מתוך מאגר הכתובות הזמינות של השרת.
- מסכת רשת: מגדירה את גודל הרשת שאליה משתייך המכשיר.
- שער ברירת מחדל: כתובת ה־IP של הנתב המשמש כברירת מחדל ליציאה מהרשת המקומית.
- שרתי **DNS**: כתובות ה־IP של שרתי ה־DNS המשמשים לתרגום שמות דומיין לכתובות IP.

ההצעה נשלחת בחזרה למכשיר החדש. המכשיר יכול לבחור לקבל את ההצעה או לדחות אותה. אם המכשיר מקבל את ההצעה, הוא שולח בקשה לשרת DHCP כדי לאשר את הבחירה.

המשך בעמוד 2

DHCP Request

כאשר המכשיר בוחר את כתובת ה־IP המוצעת על ידי שרת ה־DHCP, הוא שולח בקשה (DHCP Request) בחזרה לשרת כדי לאשר את הבחירה ולקבל את כתובת ה־IP באופן רשמי. הבקשה הזו כוללת את הפרטים הבאים:

- כתובת ה־**IP** המוצעת: המכשיר מציין את כתובת ה־IP שהוא רוצה לקבל.
- מזהה הלקוח: מזהה ייחודי של המכשיר, כמו כתובת ה־MAC שלו.
- פרמטרים נוספים (אופציונלי): המכשיר יכול לבקש פרמטרים נוספים כמו שרת DNS, שער ברירת מחדל וכו'.

הבקשה הזו מועברת לשרת ה־DHCP, והשרת בודק את הבקשה. אם הכול תקין, השרת מאשר את הבקשה ושולח אישור (DHCP ACK) למכשיר. האישור מכיל את כתובת ה־IP שהוקצתה למכשיר, יחד עם פרמטרים נוספים כמו מסכת רשת, שער ברירת מחדל ושרתי DNS.

DHCP ACK

שרת DHCP מאשר את בקשת המכשיר ושולח לו אישור DHCP (DHCP ACK). האישור כולל את הפרטים הבאים:

- כתובת ה־**IP** שהוקצתה: כתובת ה־IP שהשרת אישר למכשיר.
- מסכת רשת: מגדירה את גודל הרשת שאליה משתייך המכשיר.
- שער ברירת מחדל: כתובת ה־IP של הנתב המשמש כברירת מחדל ליציאה מהרשת המקומית.
- שרתי **DNS**: כתובות ה־IP של שרתי ה־DNS המשמשים לתרגום שמות דומיין לכתובות IP.
- זמן חיים (**Lease time**): משך הזמן שהמכשיר רשאי להשתמש בכתובת ה־IP שהוקצתה לו.

לאחר קבלת האישור, המכשיר יכול להתחיל להשתמש בכתובת ה־IP החדשה שלו ולתקשר עם מכשירים אחרים ברשת.

הגדרת נתבים ומתגים

- בכל הפקודות שם הנתב (Router Name) הוא: R1.
  שם המתג (Switch Name) הוא: S1.
- הפרמטרים שעליהם הפקודה מתבצעת/מתייחסת יופיעו בתוך < > .
  || משמעותו – או (or).

פקודות בסיסיות בנתב/במתג

הגדרת שם לנתב/מתג:	R1 or S1(config)# hostname <device_name>
סיסמה מוצפנת למצב הרשאה:	R1 or S1 (config)# enable secret <password>
כרזת כניסה:	R1 or S1 (config)# banner motd <delimiter><message><delimiter>
שמירת קובץ התצורה הקיים ב־RAM ל־NVRAM:	R1 or S1#copy run st

הגדרת סיסמה ל־console בנתב/מתג:

R1 or S1 (config)# line console <line_number>

R1 or S1 (config-line)# password <password>

R1 or S1 (config-line)# login

R1 or S1 (config-line)# login local

המשך בעמוד 3

הגדרת סיסמה להתחברות מרחוק לנתב/מתג:

```
R1 or S1 (config)# line vty <start_line> <end_line>
R1 or S1 (config-line)# password <password>
R1 or S1 (config-line)# login
R1 or S1 (config-line)# login local
```

הצפנת כל הסיסמאות בנתב או במתג:

```
R1  or S1 (config)# service password-encryption
```

פקודות show

הצגת תוכן קובץ התצורה בזיכרון RAM:

בנתב
```
R1# show running-config
```

במתג
```
S1# show running-config
```

הצגת תוכן התצורה בזיכרון NVRAM:

בנתב
```
R1# show startup-config
```

במתג
```
S1# show startup-config
```

הצגת כתובות MAC במתג:
```
S1#show mac address-table
```

הצגת תוכן טבלת הניתוב בנתב:
```
R1#show ip route
```

הצגת סיכום של כתובות ה־IP ומצב הממשקים בנתב:
```
R1# show ip interface brief
```

הצגת פרטים סטטיסטיים של ממשק:
```
R1# show interface<interface_type><number>
```

הצגת טבלת ARP:
```
R1# show arp
```

טבלת ARP במערכת הפעלה:
```
c:\>arp -a
```

המשך בעמוד 4

פקודות המתייחסות למתג (Switch)

הגדרות VLAN / TRUNK במתג:

S1(config)# vlan <vlan_id>

S1(config-vlan)# name <vlan_name>

S1(config)# interface <interface_type><interface_number>

S1(config-if)# switchport mode <access||trunk>

S1(config-if)# switchport access vlan <vlan_id>

S1(config-if)# switchport trunk allowed vlan <vlan_list>

S1# show vlan brief

S1# show vlan

הגדרות אבטחה במתג:

S1(config-if)# switchport port-security

S1(config-if)# switchport port-security maximum <number>

S1(config-if)# switchport port-security mac-address <mac_address>

S1(config-if)# switchport port-security violation <shutdown/restrict/protect>

הגדרת טווח ממשקים במתג:

Switch(config)# interface range <interface_type><start_number> - <end_number>

Switch(config)# interface range <interface_type><interface_number> , <interface_type><interface_number>

פקודות המתייחסות לנתב (Router)

הגדרת יציאה/ממשק (interface) בנתב:

R1(config)# interface <interface_type><number>

R1 (config-if)# ip address <ip_address> <subnet_mask>

R1 (config-if)# no shutdown

R1 (config-if)# description <text> פקודה אופציונלית

ביטול חיפוש ה־DNS האוטומטי בנתב:

R1(config)# no ip domain-lookup

הגדרות פרוטוקולי ניתוב

<u>הגדרת פרוטוקול OSPF</u>:

R1(config)# router ospf <process_id>

R1(config-router)# network <network_address> <wildcard_mask> area <area_number>

המשך בעמוד 5

הגדרת פרוטוקול EIGRP:

R1(config)# router eigrp <AS-number> AS= Autonomous System

R1(config-router)# network <network-address> <wildcard-mask>

הגדרת פרוטוקול RIP:

R1(config)# router rip

R1(config-router)# version <1||2>

R1(config-router)# network <network_address>

ניתוב סטטי (צף):

R1(config)# ip route <destination_network> <subnet_mask> <next_hop||ip/exit_interface>

הגדרת DHCP בנתב:

R1(config)# ip dhcp excluded-address <start_ip> <end_ip>

R1(config)# ip _____ <pool_name>

R1(config-dhcp)# network <network_address> <subnet_mask>

R1(config-dhcp)# default-router <gateway_ip_address>

R1(config-dhcp)# dns-server <dns_ip address>

R1(config-dhcp)# domain-name <domain>

R1(config-dhcp)# lease <days> [hours] [minutes]

לקבלת כתובות IP ממקטע אחר בנתב (DHCP Relay Agent): <R1(config-if)#ip helper-address <ip address

הגדרת Access Control List (ACL) בנתב:

Standard = group number 1-99

R1(config)# access-list <group_number> <permit/deny> <source_ip> <wildcard_mask>

R1(config-ext-nacl)# access-list <group-number> <permit/deny> any any

Extended = group number 100-199

R1(config-ext-nacl)# <permit/deny> <protocol> <source_ip> <wildcard_mask> <destination_ip> …

<wildcard_mask> <eq=port number>

R1(config)# ip access-list extended <name>

שיוך ליציאה: R1(config-if)# ip access-group <acl number||name> in||out

המשך בעמוד 6

הגדרת NAT בנתב:

R1(config)#interface <interface_type><number> inside interface

R1(config-if)#ip address <interface_type><number>

R1(config-if)#ip nat inside

R1(config)#interface <interface_type><number> outside interface

R1(config-if)#ip address <interface_type><number>

R1(config-if)#ip nat outside

R1(config)#access-list <group_number> permit <source ip address> <wildcard mask>

R1(config)#ip nat inside source list <group number> interface <outside interface> overload

R1(config)# ip nat inside source static <local_ip> <global_ip>

בחירת כתובות IP ציבוריות מתוך מאגר כתובות. IP ACL (pool) יקבע אילו התקנים יקבלו כתובות אלו:

R1(config)# ip nat inside source list <acl_number> pool <pool_name>

יצירת מאגר כתובות IP ציבוריות:

R1(config)# ip nat pool <pool_name> <start_ip> <end_ip> netmask <netmask>

R1(config-if)# ip nat <inside/outside>

בהצלחה!

נספח ה': מילון מונחים

לשאלון 714913, אביב תשפ"ו

המונח	תרגום המונח		
	ערבית	רוסית	אנגלית
אלגוריתם סטטי	خوارزميّة ثابتة	Статический алгоритм	static algorithm
אתחול	إبتداء	Загрузка/инициализация	initialization
בלוק	كتلة	Блок	block
גיבוב	مزيج	Хеширование	hashing
החלפת דפים	إستبدال صفحات	Замена страницы	page replacement
היררכיה	هرميّة	Иерархия	hierarchy
הקצאה	تخصيص	Назначение	allocation
הרשאת גישה	إذْن وصول	Доступ	access permission
השהיה	توقُّف مؤقّت	Пауза	pause
זיכרון מטמון	ذاكرة التخزين المؤقّت	Кэш-память	cache memory
חיפוש בינארי	بحث ثنائيّ	Двоичный поиск	binary search
חיפוש סדרתי	بحث تسلسليّ	Серийный поиск	sequential search
חריג	شاذّ	Аномальный	exception
טעינה	تحميل	Загрузка	loading
יחס החטאה	نسبة الإخطاء	коэффициент кэш-промахов	miss ratio
כשל דף	فشل الصفحة	Отказ в доступе к странице	page fault
מחיצה	فاصل	раздел	partition
מטריצה	مصفوفة	Матрица	matrix
מיפוי	رسم الخرائط	Картирование	mapping
מניעה הדדית	إقصاء مُتبادَل	Взаимное исключение	mutual exclusion
מערכת מבוזרת	نظام مُوَزَّع	Децентрализованная система	distributed system
סנכרון	تَزَامُن	Синхронизация	synchronization
פסיקה	قَطْع	Прерывание	interrupt
קובץ	مِلَفّ	Файл	file
קטע קוד קריטי	مقطع الكود الحرج	Критический участок кода	critical code section
קיפאון	ركود	Застой	deadlock
שיטת זיכרון מדומה	محاكاة نظام الذاكرة	Система виртуальной памяти	virtual memory method
תהליך אצווה	مُعالَجة بِدُفْعات	Пакетная обработка	batch process
תזמון	توقيت	Синхронизация	timing