שאלה 4

נדרש בעיקר את נוסחאות הנסיגה.
עם הוכחת נכונות, סיבוכיות והיכן נמצא

3.1 בעיית תת־המחרוזת המשותפת הארוכה ביותר LCS

קלט: 2 מחרוזות $X = x_1, \ldots, x_n$ ו־$Y = y_1, \ldots, y_m$.

מחפשים את תת־המחרוזת המשותפת (אפשר עם דילוגים), אבל לשמור על הסדר) באורך מירבי.

לדוגמא: $Y = bacbc$, $X = abccba$, אז abc תת־מחרוזת משותפת.

$opt(i,j)$ — אורך תת המחרוזת המשותפת הארוכה ביותר בין $x_1 \ldots x_i$ לבין $y_1 \ldots y_j$

$$opt(i,j) = \begin{cases} 0 & i=0 \text{ או } j=0 \\ 1 + opt(i-1, j-1) & x_i = y_i \\ \max(opt(i-1,j), opt(i,j-1)) & \text{אחרת} \end{cases}$$

סיבוכיות — $O(n^2)$

הפתרון — $opt(n,m)$

שאלה 5

מופע: שלוש סדרות של אותיות $X = x_1, x_2, \dots, x_r; Y = y_1, y_2, \dots, y_m; Z = z_1, z_2, \dots, z_n$.
נסחו אלגוריתם המשתמש ב-Memoization למציאת תת סדרה ארוכה ביותר המשותפת לשלוש הסדרות.

שאלה 6

נגדיר את הבעיה הבאה.

מופע: טקסט מקור המיוצג על ידי מערך x[1…m] של אותיות.
טקסט יעד המיוצג על ידי מערך y[1…n] של אותיות.

מטרה כללית:
להפוך את טקסט המקור לטקסט היעד על ידי סדרת פעולות עריכה אשר אותן ניתן לבחור מתפריט הפעולות שיפורט בהמשך.

לצורך העריכה הנ"ל נשתמש במערך עזר z אשר בו יוחזקו תוצאות הביניים. בהתחלה z הינו ריק ובסוף z[j]=y[j] לכל j=1…n . בכל שלב אינדקס i מצביע למקום הנוכחי הנערך כעת במערך x ואינדקס j מצביע למקום נוכחי במערך z. **הפעולות יכולות לשנות את z[j]** .
להלן תפריט הפעולות האפשריות:

Operation	Result	Cost
Copy	z[j]←x[i] , i++,j++	-1
Delete	i++	2
Replace(c)	z[j]←c, i++ , j++	1
Insert(c)	z[j]←c , j++	2

מרחק עריכה של y מ x הינו עלותה של סדרת פעולות זולה ביותר אשר הופכת את x לy .

בשאלה זו, עליכם לתאר אלגוריתם מבוסס תכנון דינאמי לחישוב מרחק עריכה של טקסט יעד מטקסט מקור.

סעיף א (5 נקודות): נסחו תת-בעיה אופיינית והגדירו את משמעותו של תא במערך המתאים.
סעיף ב (15 נקודות): נסחו נוסחת נסיגה מתאימה ותנאי בסיס.
סעיף ג (5 נקודות): נסחו משפט תת-מבנה אופטימאלי עבור הבעיה. **(ללא הוכחה)**
סעיף ד (5 נקודות): הסבירו מדוע בעת חישוב ערך מסוים כל הערכים הדרושים לשם כך כבר מחושבים. בנוסף, ספקו זמן ריצה של אלגוריתם איטרטיבי לפתרון הבעיה.
סעיף ה (10 נקודות): יהי **A** המערך אשר חושב על ידי האלגוריתם שפורט חלקית בסעיף ד. תארו כיצד יש לחשב העמדה (יישור, התאמה, **alignment**) לטקסט היעד והמקור על סמך **A** שחושב.

דוגמאות להמחשה:
טקסט מקור: **pesach** טקסט יעד: **sameach** . להלן סדרת פעולות עריכה אפשרית:

Operation	X	Z
We start here →	Pesach (i=1, on P)	(Empty, j =1)
Replace (s)	Pesach (i=2, on e)	S (j=2, location after s)
Insert (a)	Pesach (i=2, on e)	Sa (j=3, after a)
Insert(m)	Pesach (i=2, on e)	Sam (j=4 after m)
Copy	Pesach (i=3, on s)	Same (j=5 after e)
Delete	Pesach (i=4, on a)	Same (j=5 after e)
Copy	Pesach (i=5, on c)	Samea (j=6 after a)
Copy	Pesach (i=6, on h)	Sameac (j=7 after c)
Copy	Pesach (i=7, after h)	Sameach (j=8 after h)

עלות הסדרה הינה: 3.
ההתאמה בין טקסט המקור וטקסט היעד לפי הדוגמא לעיל הינה:

```
P _ _ e s ach
S a m e _ ach
```

Opt(i,j,k) – מס' הרצאות מקסימלי שאפשר לשבץ מהקבוצה: $l_1,...,l_i$ כאשר באולם הראשון משובצות הרצאות במשך j שעות, ובאולם השני הרצאות במשך k שעות.

שאלה 7

בעקבות החגיגות על ניצחונה של הפועל באר שבע השבוע, נדחה הכנס הבינלאומי שתוכנן. הפעם דאגה המחלקה למדעי המחשב לבקש את שני האודיטוריומים בבניין 26 (5 ו-6). להזכירכם סיגלית וזהבה רוצות לשבץ באותו היום מספר גדול ככל האפשר של הרצאות, מבין n הרצאות אפשריות, **<u>בשני האודיטוריומים</u>**. משך ההרצאה ה-i הוא l_i ($l_i \in N$) שעות. אין הגבלה על זמני ההתחלה והסיום של ההרצאות. יורם מסר למזכירות כי באותו יום ניתן להשתמש במשך T ($T \in N$) שעות <u>בכל</u> אחד משני האודיטוריומים. בעקבות ההצלחה בפעם הקודמת, התבקשתם לעזור לסיגלית וזהבה לשבץ **ללא התנגשויות** <u>מספר</u> מקסימלי של הרצאות בזמן T <u>בשני האודיטוריומים</u>. כאשר מתחילים הרצאה מסוימת, חייבים לסיימה (באופן רציף).

$$Opt(i,j,k) = \max\begin{pmatrix} opt(i-1,j,k), \\ opt(i-1,j-l_i,k), \\ opt(i-1,j,k-l_i) \end{pmatrix}$$

א. כתבו (שוב) את התיאור הפורמלי לבעיה (עם השינויים הנחוצים).

ב. תכננו אלגוריתם מבוסס תכנון דינמי לפתרון הבעיה עם שני אולמות:

i. נסחו והוכיחו את תכונת תת מבנה אופטימלי.

ii. נסחו את נוסחת הרקורסיה (הקפידו לנסח את משמעות תוכן תא במטריצה במילים).

iii. רשמו את תנאי הקצה עבור נוסחת הרקורסיה מהסעיף הקודם. כלומר את תתי-הבעיות שפתרונן האופטימלי ידוע מראש.

iv. תארו את הלולאה המרכזית באלגוריתם איטרטיבי מבוסס תכנון דינמי לפתרון הבעיה, כלומר ללא הלולאות המאתחלות את תנאי הקצה שפרטתם בסעיף הקודם. הסבירו מדוע בעת חישוב ערך של תא במטריצה, כל הערכים הדרושים לחישוב תא זה כבר חושבו. (עבור חצי מהניקוד על הסעיף אפשר לכתוב אלגוריתם רקורסיבי המשתמש בשיטת התזכור-Memoization).

מס' הרצאות מרבי:

```
for j=1...T
    for k=1...T
        if (opt(n,j,k) > max)
            max = opt(n,j,k)
return max
```

```
for i=1...n
    for j=0...T
        for k=0...T
            opt(i,j,k)
```

המט' מתמלא מהתחלה לסוף
בכל קריאה הערכים הקודמים
כבר קיימים

סיבוכיות – $O(n^3)$

שאלה 8

- על חישוק (ההיקף של עיגול) ממקמים n חרוזים מחמישה צבעים שונים, ובנוסף חרוז יחיד הצבוע בשחור (המיקומים נקבעים לפי כוכבים ומזלות). החרוזים ממוספרים בתחום $1..n$, כך שהחרוז השחור מצוי בין חרוז 1 לבין חרוז n.
- מחברים בין זוגות של החרוזים מצבעים זהים על ידי חוטים, כך ששני חוטים שונים (שהם שני מיתרים בעיגול) לא יחצו זה את זה, ולכל חרוז מחובר לכל היותר חוט אחד.
- כדי שלוכד החלומות יהיה בעל יעילות מרבית יש למתוח מספר מקסימלי של חוטים (מיתרים).

<u>בעיית לוכדי החלומות</u>: בהינתן חישוק אשר ממוקמים עליו n חרוזים כמתואר, כיצד ניתן לחבר את החרוזים ע"י חוטים כך שלוכד החלומות יהיה בעל יעילות מרבית?

א) ניתן לפתור בעיה זו בתכנון דינמי. הסבר/י מדוע, ונסח/י את הטענה המתאימה עבור הבעיה הזאת.

ב) הגדר/י מה מיצג תא בטבלה שתמולא במהלך האלגוריתם.

ג) תן את נוסחת רקורסיה מתאימה.

ד) תאר/י (עדיף בפסאודו-קוד בעברית/אנגלית) גרסה איטרטיבית לאלגוריתם, הרץ בזמן $O(n^3)$. עבור מחצית מהניקוד על סעיף זה, ניתן להציג גרסה לא איטרטיבית הרצה בזמן זהה.

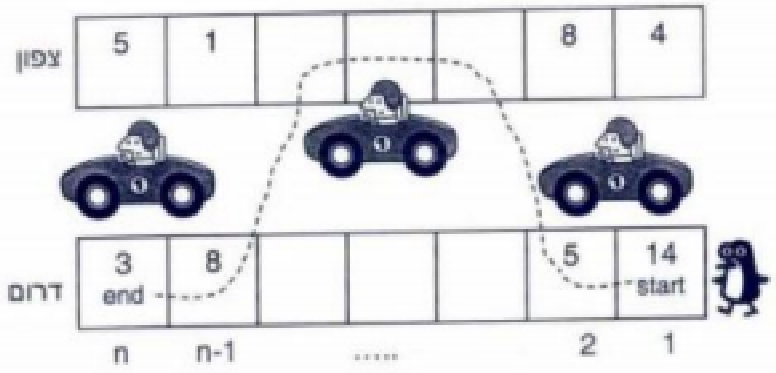
ה) האם הפתרון היה שונה, אילו היו על החישוק רק n הקדקודים הצבעוניים (ללא החרוז השחור)? הסבר/י.

שאלה 9

התבקשת לייעץ למנהל חברה אשר מתכנן מסיבה לעובדיו. לחברה יש מבנה היררכי שבו המנהל הוא שורש העץ. מחלקת כוח אדם נתנה לכל עובד ציון ב-"עליזות" – מספר ממשי. כדי שהמסיבה תהיה כיפית, המנהל מבקש שלא יוזמנו אליה גם עובד וגם הבוס הישיר שלו. כתוב אלגוריתם אשר יניב רשימת אורחים כזו שסכום ציוני ה-"עליזות" של כל המוזמנים יהיה המירבי. נתח את זמן הריצה של האלגוריתם.

שאלה 10

נתון כביש שמשני צדדיו מדרכות משובצות. על כל משבצת מונחת כמות מסוימת של גרגירי תירס. פינגווין רעב מגיע למשבצת הראשונה בצד הדרומי של הכביש. בכל פעם שהפינגווין דורך במשבצת מסוימת הוא אוכל את כל גרגירי התירס שעליה.



הפינגווין מעוניין לאכול כמה שיותר גרגירי תירס בדרכו מהמשבצת הראשונה למשבצת ה-n-ית. בכל צעד הוא יכול להתקדם משבצת אחת קדימה (מערבה) באותו צד, או לחצות את הכביש (מצפון לדרום או להיפך) תוך התקדמות משבצת אחת קדימה (מערבה).

במהלך הטיול מותר לפינגווין לחצות את הכביש $d<n$ פעמים לכל היותר. עליו לסיים את הטיול במשבצת המערבית (ה- n-ית) הדרומית. הקו המרוסק בציור מתאר מסלול אפשרי עם שתי חציות.
נסמן ב-a_i את מספר גרגירי התירס במשבצת ה-i-ית בצד בדרומי וב-b_i את מספר גרגירי התירס במשבצת ה-i-ית בצד בצפוני. (לכל i בין 1 ל-n). למשל בציור $a_2=5$, $b_1=4$. ערכי a_i ו-b_i ידועים מראש לכל i.

בשאלה זו עליכם לתאר אלגוריתם המבוסס על **תכנות דינאמי** העוזר לפינגווין לתכנן את מסלולו כך שבסך הכל יאכל במהלך הטיול את הכמות המירבית האפשרית של גרגירי תירס. נסתפק בחישוב הכמות ואין צורך לכלול בפלט כיצד להשיגו.

נסמן ב-$S(i,k)$ את כמות הגרגרים המירבית שניתן לצבור עד (וכולל) המשבצת ה-i–ית אם בוצעו בדיוק k חציות עד משבצת זו. שימו לב שלערכי $S(i,k)$ יש משמעות רק עבור $i>k$.

א. (3 נק') נניח שערכי $S(i,k)$ ידועים לכל $i>k$ ו- $k\leq d$. מהי הכמות המירבית האפשרית של גרגירי תירס שיכול הפינגווין לאסוף (כביטוי התלוי בערכים אלו)?

ב. (12 נק') כתבו נוסחת תכנות דינאמי לחישוב $S(i,k)$, הפרידו בין מקרי הבסיס והנוסחא הכללית.

ג. (5 נק') תארו במילים מהי הטבלה אותה ימלא אלגוריתם התכנות הדינאמי, מהו סדר מילוי הערכים בטבלה, ומהי סיבוכיות הזמן של האלגוריתם.

יש להסביר ולנמק את התשובות!

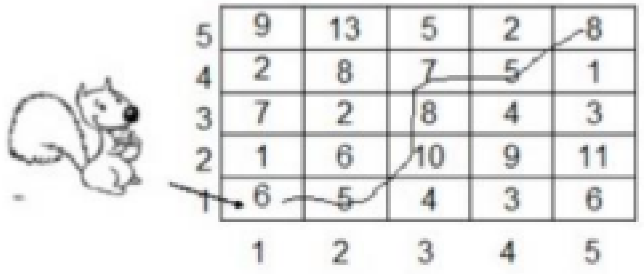
שאלה 11

נתון חדר רבוע בעל $N*N$ מרצפות. על כל מרצפת (i,j) מונחת כמות נתונה $p(i,j)$ של אגוזים. סנאי רעב נמצא במרצפת $(1,1)$ ומעוניין להגיע למרצפת (N,N). הסנאי יכול לאסוף בדרכו את כל האגוזים שנמצאים על המרצפות בהן הוא עובר.
הבהרה: באיור, מרצפת (i,j) נמצאת בעמודה ה-i משמאל ובשורה ה-j מלמטה, למשל $p(4,2)=9$.

ממרצפת (i,j) הסנאי יכול להתקדם בצעדים הבאים בלבד:
1. קפיצה מאונכת - למרצפת $(i, j+1)$ [מותרת עבור $i\le N,\ j<N$]
2. קפיצה מאוזנת - למרצפת $(i+1,j)$ [מותרת עבור $i<N,\ j\le N$]
3. קפיצה אלכסונית - למרצפת $(i+1,j+1)$ [מותרת עבור $i<N,\ j<N$]

קל לראות שקפיצה אלכסונית אינה כדאית, כי הסנאי יאסוף יותר אגוזים אם יחליף קפיצה אלכסונית בשתי קפיצות (ואז יאסוף בנוסף למה שנמצא ב-$(i+1,j+1)$ את מה שנמצא ב-$(i+1,j)$ או ב-$(i,j+1)$).
עם זאת, מאחר שהסנאי ממהר, הוא <u>חייב לבצע D צעדים אלכסוניים</u>, עבור $0<D<N$ נתון.

באיור מתואר מסלול אפשרי שהסנאי אוסף בו $6+5+10+8+7+5+8$ אגוזים. במסלול זה בוצעו שני צעדים אלכסוניים.



בשאלה זו נעזור לסנאי למצוא, בעזרת **תכנות דינאמי**, מסלול שכולל D צעדים אלכסוניים ומאפשר לו לאסוף בדרכו מספר אגוזים מקסימלי.
עבור ערכי $p(i,j)$ נתונים, נסמן ב $M_h(i,j)$ את המספר המקסימלי של אגוזים שניתן לאסוף במסלול ממרצפת $(1,1)$ למרצפת (i,j) שיש בו בדיוק h צעדים אלכסוניים. ערכי M מוגדרים עבור חלק מהערכים $1\le j\le N$, $1\le i\le N$, $0\le h\le D$. ל-$M_2(1,3)$ למשל, אין משמעות, מכיוון שלא ניתן להגיע ל-$(1,3)$ בשום מסלול חוקי שיש בו שני צעדים אלכסונים.

א. (2 נק') מהו הביטוי אותו מעוניינים להביא לערך מקסימלי?
ב. (11 נק') כתבו נוסחת תכנות דינאמי לחישוב $M_h(i,j)$. הפרידו בין הבסיס למקרה הכללי.
ג. (4 נק') תאח במילים מהי הטבלה אותה ימלא אלגוריתם התכנות הדינאמי, מהו סדר מילוי הערכים בטבלה, ומהי סיבוכיות הזמן של האלגוריתם.
ד. (3 נק') כיצד ניתן למצוא את המסלול האופטימלי? תשובתם יכולה להתבסס על הטבלה וערכי $p(i,j)$, או על תוספת לחישוב $M_h(i,j)$.

יש להסביר ולנמק את התשובות!

שאלה 12

נניח שבסימסטר מסוים אתם לומדים **n** קורסים. לקראת סיום הסימסטר, יש לבצע פרוייקט בכל אחד מ-n הקורסים. הציון על כל פרוייקט הוא מספר בין 0 ל-100.
ברשותכם H שעות אותן בכוונתכם להשקיע בביצוע הפרויקטים. נניח שלכל אחד מהפרויקטים $1 \le i \le n$ ידועה לכם פונקציה $f_i:\{0,\dots,H\} \to \{0,\dots,100\}$ כאשר $f_i(h)$ הוא הציון שתקבלו בפרוייקט ה-i אם תשקיעו בו h שעות. ידוע שלכל i הפונקציה f_i אינה יורדת ומוגדרת רק עבור מספרים שלמים (ניתן להניח ש-H שלם ושעליכם להשקיע מספר שעות שלם בכל פרוייקט).
למשל אם עבור הפרוייקט השני $f_2(0)=10$, $f_2(1)=30$, $f_2(2)=40$, $f_2(3)=70$ ולכל $4 \le k \le H$ $f_2(k)=85$, אז אם לא תשקיעו כלל זמן תקבלו 10, אם תשקיעו שעה אחת תקבלו 30, אם תשקיעו שעתיים תקבלו 40, אם תשקיעו 3 שעות תקבלו 70, ואם תשקיעו 4 או יותר שעות תקבלו 85.
המטרה שלכם היא כמובן להחליט כיצד לחלק את H השעות שברשותכם כך שסכום הציונים שתקבלו בכל הפרויקטים יחד יהיה גבוה ביותר.

א. סטודנט מסוים הציע את האלגוריתם הגרידי הבא לפתרון הבעיה:
לכל פרוייקט, המשתנה n_i יציין כמה שעות כבר הוקצו לפרוייקט ה-i-י.

```
Init: for i=1 to n,  n_i=0
Loop: For j=1 to H
          Let i be the project for which f_i(n_i+1)-f_i(n_i) is maximal.
               n_i = n_i+1.
```

1. (3 נק') השלימו במשפט אחד, במילים בלבד,(בלי להשתמש בשמות המשתנים או ב-f) מה עושה האלגוריתם.

2. (8 נק') האם האלגוריתם אופטימלי? הוכיחו או תנו דוגמא נגדית.

ב. בחלק זה עליכם לתאר אלגוריתם לפתרון הבעיה בעזרת **תכנות דינאמי**. נסתפק בחישוב הסכום המירבי ואין צורך לכלול בפלט כיצד להשיגו (כמה שעות להקדיש לכל פרויקט).

לכל $1 \le j \le n$, $0 \le k \le H$, נסמן ב-$D[j,k]$ את סכום הציונים המקסימלי ב-j הקורסים הראשונים, אם תקדישו להם (יחד) k שעות.

א. (2 נק') מהו הביטוי אותו מעוניינים להביא לערך מקסימלי?
ב. (13 נק') כתבו נוסחת תכנות דינאמי לחישוב $D[j,k]$.הפרידו בין הבסיס למקרה הכללי.
ג. (5 נק') תארו במילים מהי הטבלה אותה ימלא אלגוריתם התכנות הדינאמי, מהו סדר מילוי הערכים בטבלה, ומהי סיבוכיות הזמן של האלגוריתם.

יש להסביר ולנמק את תשובותכם.

שאלה 13

לוויין Google Earth נשלח לגיחת צילום. הלוויין מסוגל לצלם 2 סוגי תמונות – תמונה ברזולוציה גבוהה (HD) ותמונה ברזולוציה רגילה (SD). בידיכם רשימה של n אתרים אפשריים לצילום, נסמנם $\{1,2,\dots,n\}$. לכל אתר נקבע רווח מצילום ברזולוציה גבוהה $\{h_1,h_2,\dots,h_n\}$ ורווח מצילום ברזולוציה רגילה $\{s_1,s_2,\dots,s_n\}$ בהתאמה. עליכם לתכנן גיחת צילום רווחית ככל הניתן בזמן קצוב.

שימו לב: לכל $1 \le i,j \le n$ מעבר הלוויין מאתר i לאתר j **אינו** דורש זמן.

עבור כל אתר i, יש ללוויין **__בדיוק אחת__** משלוש האפשרויות הבאות:

1. לא לצלם את האתר ה-i.
2. לצלם את האתר ה-i ברזולוציה רגילה. משך הפעולה: **3 דק'**.
3. לצלם את האתר ה-i ברזולוציה גבוהה. משך הפעולה: **5 דק'**.

הלוויין עומד לרשותכם למשך T דקות (מספר טבעי כלשהו, ניתן להניח כי $T \le 5n$).

תכנית צילום חוקית מוגדרת כשתי תת-קבוצות **זרות** $H,S \subseteq \{1,\dots,n\}$, $H \cap S = \emptyset$ ומתקיים - $5|H| + 3|S| \le T$.

שווי תכנית צילום (H,S) מוגדר כ- $\sum_{i\in H} h_i + \sum_{i\in S} s_i$.

בסעיפים א'-ג' נשתמש בתכנון דינאמי על מנת לחשב את __השווי המקסימלי__ של תכנית צילום חוקית כלשהי:

סעיף א (5 נקודות)

נסחו (במילים) מהי תת-בעיה אופיינית ע"י הגדרת ערך OPT מתאים __ללא הנוסחה המתאימה עבור ערך OPT זה__.

סעיף ב (7 נקודות)

תארו נוסחת מבנה הכוללת מקרה/י בסיס והסבירו **בקצרה** את נכונות הנוסחה.

סעיף ג (7 נקודות)

נסחו אלגוריתם איטרטיבי (יעיל ככל הניתן) למציאת **__שווי תוכנית צילום__** אופטימאלית, ונתחו את זמן ריצתו. אין צורך בהוכחת נכונות. על האלגוריתם לאתחל ולמלא מבנה נתונים מתאים.

בסעיף הבא נשחזר תוכנית צילום אופטימאלית:

סעיף ד (6 נקודות)

נסחו אלגוריתם לשחזור **__תוכנית צילום אופטימאלית__** (זיכרו כי יש להגדיר מתוך הקבוצות H,S) המבוסס על מבנה נתונים נתון אשר נבנה באלגוריתם מסעיף ג', ונתחו את זמן ריצתו. אין צורך בהוכחת נכונות.

שאלה 14

<u>מופע:</u> זוג סדרות של מספרים שלמים חיוביים: $P=\{p_1,\dots,p_m\}$ ו $H=\{h_1,\dots,h_n\}$ כך ש $n\le m$.
ניתן להניח ש H ממוינת בסדר עולה.

<u>פתרון חוקי:</u> סדרה $\{i_1,i_2,\dots,i_n\}\subseteq[1,m]$ של אינדקסים של איברים ב P.

<u>יש למצוא:</u> פתרון חוקי $M=\{i_1,i_2,\dots,i_n\}$ כך ש $d(M)=\sum_{k=1}^{n}|p_{i_k}-h_k|$ מינימאלי.

סעיף א: הראו כי האלגוריתם החמדן הבא נכשל:

1. מיין את P בסדר עולה.
2. החזר $\{i_1,i_2,\dots,i_n\}$ האינדקסים של איברי P שמתאימים ל n האיברים הראשונים לפי המיון.

סעיף ד: הגדירו תת-בעיה אופיינית. כלומר הגדירו $OPT(i,j)$ כאשר i מייצג את P ו j את H. ציינו איזה ערך אנו מעוניינים לחשב.

סעיף ה: נסחו נוסחת נסיגה ותנאי בסיס עבור ערך של פתרון אופטימאלי עבור הבעיה לעיל.
<u>הדרכה:</u> עבור $OPT(i,j)$ שהגדרתם שקלו שני מקרים: $i\ge j$, ו $i<j$.

15 שאלה

Problem 4. Canoe Rental Problem:
There are n trading posts numbered 1 to n as you travel downstream. At any trading post i you can rent a canoe to be returned at any of the downstream trading posts $j > i$. You are given a cost array $R(i, j)$ giving the cost of these rentals for all $1 \leq i < j \leq n$. We can assume that $R(i, i) = 0$, and that you can't go upriver (so perhaps $R(i, j) = \infty$ if $i > j$). For example, one cost array with $n = 4$ might be the following.

C		to 1	j 2	3	4
from i	1	0	2	3	7
	2	–	0	2	4
	3	–	–	0	2
	4	–	–	–	0

The problem is to find a dynamic programming algorithm that computes the cheapest sequence of rentals taking you from post 1 all the way down to post n. In this example, the cheapest way is to rent canoes from post 1 to post 3, and then from post 3 to post 4 for a total cost of 5. The second problem is to find the least cost associated with this sequence.

You are to design a dynamic programming algorithm for both the problems. Describe the table and what does each entry in the table mean? How will the table be initialized? In which order the table will be filled? What is the recurrence? How will you use the table to find what is the cheapest sequence of canoe rentals (for the first problem) and the least cost of the canoe rentals (for the second problem)? Compute the asymptotic complexity of the algorithms. It is very important that you practice writing your own solutions to this problem even though you may have perfect understanding of the solution.

שאלה 16

Given a sequence n positive integers, $k_1, k_2, \ldots, k_n$, that sum to s (you can assume that s is even), find a subset I of $\{1, \ldots, n\}$ such that

$$\sum_{i \in I} k_i = \sum_{i \notin I} k_i = s/2$$

or determine that there is no such subset.

You are to find a dynamic programming solution to this problem.

שאלה 17

9. *Shortest path counting* A chess rook can move horizontally or vertically to any square in the same row or in the same column of a chessboard. Find the number of shortest paths by which a rook can move from one corner of a chessboard to the diagonally opposite corner [Gar78], p.10

(a) by a dynamic programming algorithm.

(b) by using elementary combinatorics.

שאלה 18

▷ *World Series odds* Consider two teams, A and B, playing a series of games until one of the teams wins n games. Assume that the probability of A winning a game is the same for each game and equal to p and the probability of A losing a game is $q = 1 - p$. (Hence, there are no ties.) Let $P(i, j)$ be the probability of A winning the series if A needs i more games to win the series and B needs j more games to win the series.

a. Set up a recurrence relation for $P(i, j)$ that can be used by a dynamic programming algorithm.

b. Find the probability of team A winning a seven-game series if the probability of it winning a game is 0.4.

c. Write a pseudocode of the dynamic programming algorithm for solving this problem and determine its time and space efficiencies.

שאלה 19

▷ Design a dynamic programming algorithm for the ***change-making problem***: given an amount n and unlimited quantities of coins of each of the denominations $d_1, d_2, \ldots, d_m$, find the smallest number of coins that add up to n or indicate that the problem does not have a solution.