

ת"ז :



החוג למערכות מידע ניהול
תכנות בסיסי (בחינות סמסטר)
מועד א' סמסטר א' - תשפ"ג
מרצה: ד"ר רמי רشكובי

משך הבחינה: סה"כ 3 שעות

השימוש בכל חומר עזר כתוב מותר

הנחיות לנבחן:

- יש לענות על כל השאלות בצורה נקיה ומסודרת בכתב יד, או באמצעות המחשב.
יש להעתיק את הקוד מסביבת הפיתוח לקובץ word אותו יש להעלות לאתר בסיום המבחן
יש לסמן במפורש קטיעתיותה, אחרת אלו ייבדקו וייחשבו כחלק מהפתרונות
אין להעביר חומר מסטודנט לסטודנט במהלך הבחינה או לדבר עם סטודנטים אחרים
שים לב לקיום העמודים בטופס הבחינה ולרציפות מספריהם
קראו היטב את השאלות והתייחסו לכל הפרטים
- .1
.2
.3
.4
.5
.6

ניקוד כולל	
/20	שאלה 1
/20	שאלה 2
/20	שאלה 3
/20	שאלה 4
/20	שאלה 5
/10	שאלה 6
/110	סיכום

בהצלחה !

הנחיות

עליכם לפתרו את 5 השאלות הבאות. כל שאלה במשקל 20 נקודות. בנוסף, יש שאלת בונוס בשווי 10 נקודות בסוף המבחן. ציון מקסימלי בבחינה - 100

שאלה 1 (20 נק')

נתונה הפונקציה הבאה :

```
def mystery(s:str)->list:
    s = s.lower()
    mylist = []
    for l in 'abcdefghijklmnopqrstuvwxyz':
        if l not in s:
            mylist.append(l)
    return mylist
```

- א. מה עושה הפונקציה? יש לתאר במשפט קצר את המהוות ולא את האלגוריתם! (15 נק')
- לדוגמא : הפונקציה מחשבת ומוחזירה את מספר התווים באנגלית של המחרוזת שהתקבלה כפרמטר
- ב. מה יודפס עבור הפוקודה הבאה ((i:rst)'abcdefghijklmnopqrstuvwxyz'? נמק בקצתה

פתרון

- א. הפונקציה מוצאת את כל התווים באנגלית (lower case) שאינם נמצאים במחרוזת שהתקבלה כפרמטר, ומוחזירה רשימה של אותיות אלו
- ב. יודפס [u,v,w,x,y,z].

שאלה 2 (20 נק')

כתבו תוכנית פיתונו אשר קולטת מהמשתמשתו כלשהו ואחריו מספר שלם חיובי כלשהו ולאחר מכן התוכנית מפעילה פונקציה מקבלת כפרמטרים את התו והמספר מדפסה למסך את המבנה הבא.
להלן שתי דוגמאות :

אם המשמש הכנס תו '@' ואחריו המספר 3 אז יודפס למסך.

@@@
@@@
@@
@@
@
@@
@@
@@@
@@@

אם המשמש הכנס תו '*' ואחריו המספר 4 אז יודפס למסך.

**

**

*

```
**
**
***
***
```

```
def myprint(char, num):
    for i in range(num, 1, -1):
        print(char * i)
        print(char * i)

    print(char)

    for i in range(2, num+1):
        print(char * i)
        print(char * i)

char = input('enter char: ')
num = int(input('enter num: '))
myprint(char, num)
```

הערה : שימו לב היטב לדוגמה ולמבנה החזרה!

שאלה 3 (20 נק')

כתבו פונקציה המקבלת כפרמטר רשימה של מספרים שלמים חיוביים, ומחזירה את המספר השכיח ביותר ברשימה (המספר שופיע הכי הרבה פעמים). לדוגמה עבור הרשימה [1,3,4,5,4,6,4,5,7] יוחזר הערך 4 שכן הוא מופיע 3 פעמים (יותר מכל מספר אחר ברשימה).שים לב: אסור להשתמש בפונקציה count של list.

```
def common(mylist:list) ->int:
    dict = {}

    for n in mylist:
        if n in dict.keys():
            dict[n] += 1
        else:
            dict[n] = 1

    v = 0
    k = 0

    for key, value in dict.items():
        if value > v:
            v = value
            k = key

    return k

print(common([1,3,4,5,4,6,4,5,7]))
```

שאלה 4 (20 נק')

יש לכתוב פונקציה המקבלת כפרמטרים שמויות של שני קבצים, פותחת את הקובץ הראשון לקרוא את השני כתיבה, וכותבת את תוכנו של קובץ הקלט אך תוך קובץ הפלט תוך מספר שורותיו באופן המוצג בדוגמה הבאה. אם תוכן קובץ הקלט נראה כך:

So, so you think you can tell
 Heaven from hell
 Blue skies from pain
 Can you tell a green field
 From a cold steel rail
 A smile from a veil
 Do you think you can tell

אז, קובץ הפלט יוכל את התוכן הבא (שים לב ל-Tab לאחר מספר השורה):

1. So, so you think you can tell
2. Heaven from hell
3. Blue skies from pain
4. Can you tell a green field
5. From a cold steel rail
6. A smile from a veil
7. Do you think you can tell

ניתן להניח ששמות הקבצים המתקבלים תקינים ושוניים זה מזה, וכיקובץ הקלט קיים.

```
def linenumbers (filename, outfilename) :
    infile = open(filename, 'r')
    lines = infile.readlines()
    infile.close()

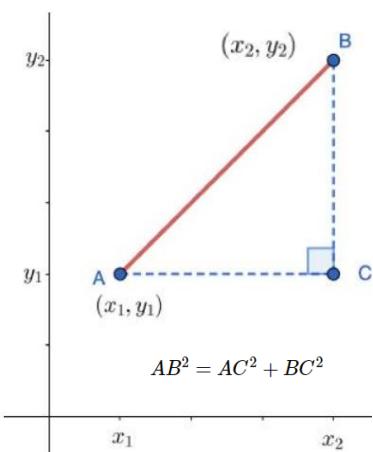
    outfile = open(outfilename, 'w')
    lineno = 1
    for line in lines:
        outfile.write(str(lineno) + '.\t' + line)
        lineno += 1
    outfile.close()

linenumbers('input.txt', 'output.txt')
```

שאלה 5 (20 נק')

כתבו פונקציה המתקבל כפרמטר רשימה של המילה קואורדינטות (y, x) של נקודות שונות במשורט.
להלן דוגמה לרשימה כזו $[(1,1), (2,3), (5,5), (6,6)]$. עליהם למצוא את המרחק הקצר ביותר בין שתי הנקודות
הקרובות ביותר ברשימה ולהחזיר אותו מהפונקציה.

רמז : כזכור מתרגיל בית 2 ניתן לחשב מרחק בין שתי נקודות באמצעות
משפט פיתגורס.



הערות :
ניתן להניח שהרשימה כוללת לפחות לפחות 2 נקודות
ניתן להוסיף פונקציות עזר למימוש

$$AB^2 = AC^2 + BC^2$$

```

import math

def calc_distance(point1, point2):
    x1, y1 = point1
    x2, y2 = point2
    return math.sqrt(math.pow(x1 - x2, 2) + math.pow(y1 - y2, 2))

def shortest_distance(coords:list):
    min_dist = calc_distance(coords[0], coords[1])
    for index1 in range(len(coords)):
        for index2 in range(index1+1, len(coords)):
            curr_dist = calc_distance(coords[index1], coords[index2])
            if curr_dist < min_dist:
                min_dist = curr_dist

    return min_dist

```

שאלה 6 - שאלה בונוס (10 נק')

כתבו פונקציה בפייתון אשר מקבלת כפרמטר רשימה של 99 ערכים ייחודיים (לא חוזרים) של מספרים. נתון שכל המספרים ברשימה הם בטוחה של 1-100, כך שחרס רק ערך אחד מתחום זה. על הפונקציה להחזיר את הערך החסר.

```

def missingValue(mylist:list)->int:
    for num in range(1, 101):
        if num not in mylist:
            return num

```

פתרון חלופי :

```

def missingValue(mylist:list)->int:
    dict = {}
    for i in range(1,101):
        dict[i] = 0

    for num in mylist:
        dict[num] = 1

    for k,v in dict.items():
        if v == 0:
            return k

```