

## הנחיות כלליות

יש לשלוח את הקבצים באמצעות [מערכת ההגשה](#) לפני חלוף השעה **23:50** בתאריך **16/11/2020**.

ניתן להגיש את התרגיל באיחור עם קנס אוטומטי על פי הפירוט הבא:

- יום איחור – קנס של **10 נקודות** (ציון מקסימלי – 90).
- יומיים איחור – קנס של **20 נקודות** (ציון מקסימלי – 80).
- שלושה ימי איחור – קנס של **30 נקודות** (ציון מקסימלי – 70).

לאחר מכן לא ניתן יהיה להגיש את התרגיל (ציון 0).

המתרגל האחראי על התרגיל הוא לידור.

שאלות בנוגע לתרגיל יש לפרסם **באופן ציבורי** בפורום הקורס בלבד. רק אם לא התקבלה תשובה לאחר 24 שעות, יש לשלוח מייל לכתובת [alislid@biu.ac.il](mailto:alislid@biu.ac.il) עם קישור לדיון (פוסט) הרלוונטי.

בקשות להארכה מסיבות מוצדקות (מילואים, לידה, אשפוז וכו') יש לפרסם **באופן פרטי** בפורום הקורס בלבד (יש למען את הפוסט ל-Instructors). בכל בקשה יש לציין: שם מלא, שם משתמש במערכת ההגשה, מספר תעודת זהות, האם אתם ממדעי המחשב או ממתמטיקה.

ש להקפיד מאוד על הוראות עיצוב הקלט והפלט, בדיוק על פי הדוגמאות המצורפות. אין להוסיף או להשמיט רווחים או תווים אחרים, ואין להחליף אותיות גדולות בקטנות או להיפך. חוסר הקפדה על פרטים אלו עלול לגרור הורדה משמעותית ביותר בציון התרגיל עד כדי 0. ראו עצמכם הוזהרתם!

**שימו** ❤️ שאתם עוקבים במדויק אחרי ההנחיות במסמך ה-Coding Style המפורסם בפורום הקורס.

עליכם לכתוב קוד על פי הוראות התרגיל ולוודא שקיבלתם 100 בבדיקה האוטומטית הראשונית, וכן שהתרגיל מתקמפל ורץ על שרתי המחלקה (u2) **ללא שגיאות וללא אזהרות**. תרגיל שלא עומד בסטנדרטים הבסיסיים הללו יגרור, בשל הטרחה שהוא מייצר בתהליך הבדיקה שלו, הורדת נקודות משמעותית בציון שלו.

להזכיר העבודה היא אישית. "עבודה משותפת" דינה כהעתקה. העתקות נבדקות על ידי מערכת ההגשה האוטומטית, ותרגיל שהועתק יגרור ציון 0 ופגיעה בציוני התרגול הסופיים **לכל הגורמים** השותפים בהעתקה. אתם יכולים לדון בגישות לפתרון התרגיל באופן תיאורטי, אך אין לשתף קוד בשום צורה.

בפיתוח הקוד ניתן להשתמש בכל סביבת עבודה, העיקר הוא שתדעו איך לקחת את קבצי הקוד מתוך הסביבה הזו, לבדוק אותם על שרתי האוניברסיטה, ולהגיש אותם באמצעות מערכת ההגשה.

דוגמאות לחלק מסביבות העבודה האפשריות:

IDEs (Integrated Development Environment):

- Visual Studio
- Clion
- Eclipse
- Xcode

Text Editors:

- Sublime Text
- Atom
- Notepad++
- Vim

בהצלחה!

## תרגיל 2 – Ex2

משקל התרגיל מתוך ציון התרגול: 10%.

בתרגיל זה עליכם ליצור ולהגיש תוכנית בקובץ יחיד בשם ex2.c. הספרייה היחידה המותרת לשימוש בתרגיל זה הינה stdio.h.

**פקודת הקומפילציה (ב-u2):** gcc \*.c -std=c99

בתחילת התוכנית יודפס למסך התפריט הבא:

Choose an option:  
 1. Rhombus  
 2. Base 20 to Decimal  
 3. Base to Decimal  
 4. Pow2  
 5. Different bits  
 6. Add  
 7. Multiply  
 0. Exit

לאחר מכן, על פי בחירת המשתמש, תתבצע המשימה המבוקשת.

בסיום ביצוע משימה, יש להדפיס מחדש את התפריט ולפעול על פי בחירת המשתמש.

במידה והמשתמש מקיש 0, יש לסיים את התוכנית.

במידה והמשתמש מקיש אופציה שאינה מופיעה בתפריט, יש להדפיס:

Wrong option!

**ולהדפיס מחדש את התפריט.**

**שימו לב:** לאחר הדפסת התפריט יש לרדת שורה

הערה כללית לכל סעיפי התרגיל: בסיום כל משימה אין להשאיר תווים לא שקופים בתוך stdin,

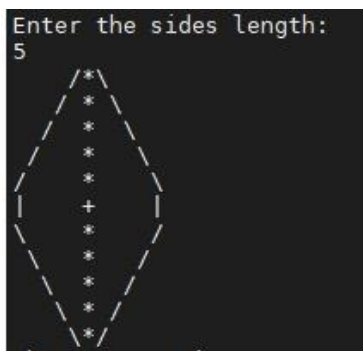
משום שזה עלול להשפיע על התפקוד של משימה שתתבצע לאחר מכן.

בסוף הקובץ מופיעה דוגמת הרצה מפורטת לכל התרגיל (שימו לב להדפסות המנחות את המשתמש שלא

מופיעות תמיד בהנחות הסעיף, למען הפשטות)

### משימה 1 – ציור מעוין

בהינתן קלט של מספר טבעי מהמשתמש לאורך צלע של המעוין יש לצייר מעוין באופן הבא:



- צלעות המעוין בגודל שקבע המשתמש (כל צלע בגודל שנקבע ע"י המשתמש – בדומה מימין – 5 ולכן כל צלע באורך 5)
- בחלק האנכי קיים אלכסון שמורכב מכוכביות (\*) ובמפגש האלכסונים קיים הסימן +.
- הפרדה במרכז המעוין בין הצלעות שנראות במאוזן על ידי סימן bitwise or ( | )
- ניתן להניח תקינות הקלט - ללא מספרים שליליים/אפס או תווים שאינם תקינים

שימו לב שהבדיקה הינה אוטומטית ולכן יש לצייר את המעוין במדויק. מעוינים "דומים" לא יתקבלו במערכת הבדיקה. דוגמאות למעוינים נוספים יימצאו בדוגמאות ההרצה.

שימו לב לא להשאיר רווחים אחרי הצלעות הימניות במעוין

### משימה 2: המרה מבסיס 20 לבסיס דצימלי

במשימה זו עליכם לקלוט מהמשתמש טקסט באורך לא ידוע (אינסופי בפוטנציאל שלו). המשתמש יקליד רצף תווים כלשהו ובסיומו יקליד אנטר (יש להתעלם מרצף אפשרי של תווים שקופים המופיע בתחילת המחרוזת).

המשתמש מקליד מספר בבסיס 20 בסדר כתיבה הפוך, כלומר – הספרה הראשונה שהוא מקליד תהיה ספרת האחדות (המטרה כאן היא להקל עליכם בהמרת המספר, נסו להבין למה זה עוזר). במידה והקלט תקין, עליכם להדפיס את ערכו של המספר בבסיס 10.

- טקסט חוקי יחשב לטקסט המכיל ספרות בבסיס 20 בלבד (שימו לב שספרות שערכן גדול מ-10 יכולות להיכתב גם באות גדולה וגם באות קטנה). במידה והטקסט מכיל תווים אחרים יש להדפיס את הודעת השגיאה:

Error! %c is not a valid digit in base 20

כאשר %c מסמן את התו הלא חוקי. לדוגמה:

Enter a reversed number in base 20:

1R1

Error! R is not a valid digit in base 20

לאחר מכן יש לחזור לתפריט הראשי.

שימו לב! בבסיס עשרים התווים החוקיים הינן הספרות 0-9 וכן האותיות:

A-10,B-11,C-12,D-13,E-14,F-15,G-16,H-17,I-18,J-19

כפי שנאמר מותר גם באותיות קטנות וגם בגדולות.

לדוגמה:

Enter a reversed number in base 20:

1aJ

7801

שימו לב לאחר הבקשה Enter a reversed number in base 20: יש ירידת שורה.

### משימה 3: המרה מבסיס כללי לדצימלי

במשימה זו עליכם לקלוט מהמשתמש מספר המייצג בסיס ספירה בין הערכים 2-10 (כולל שניהם). ניתן להניח שהמשתמש יכניס מספר בטווח הזה ושהקלט תקין. ולאחר מכן טקסט באורך לא ידוע (אינסופי בפוטנציאל שלו). המשתמש יקליד רצף תווים כלשהו ובסיומו יקליד אנטר (יש להתעלם מרצף אפשרי של תווים שקופים המופיע בתחילת המחרוזת).

המשתמש מקליד מספר בבסיס שצוין על ידו בסדר כתיבה הפוך, כלומר – הספרה הראשונה שהוא מקליד תהיה ספרת האחדות (המטרה כאן היא להקל עליכם בהמרת המספר, נסו להבין למה זה עוזר). במידה והקלט תקין, עליכם להדפיס את ערכו של המספר בבסיס 10.

- טקסט חוקי יחשב לטקסט המכיל ספרות בבסיס שהמשתמש ציין בלבד. במידה והטקסט מכיל תווים אחרים יש להדפיס את הודעת השגיאה (על פי סדר התווים של המשתמש):

```
Error! %c is not a valid digit in base %d
```

כאשר %c מסמן את התו הלא חוקי ו-%d מסמן את הבסיס הנבחר. לדוגמה:

```
Enter a base (2-10):
5
Enter a reversed number in base 5:
567
Error! 5 is not a valid digit in base 5
Error! 6 is not a valid digit in base 5
Error! 7 is not a valid digit in base 5
```

דוגמה לקלט תקין:

```
Enter a base (2-10):
2
Enter a reversed number in base 2:1101
11
```

שימו לב לאחר הבקשה `Enter a base (2-10):` יש ירידת שורה.

#### **משימה 4: בדיקה האם מספר הוא חזקה שלמה של 2**

בהינתן קלט מהמשתמש של **מספר טבעי** (ניתן להניח תקינות הקלט שכן לא יוכנסו מספרים שליליים) עליכם להחזיר אם המספר הוא חזקה שלמה של 2 או שאינו חזקה שלמה של 2.

דוגמאות:

```
Enter a number:
1024
1024 is a power of 2
```

```
Enter a number:
3
3 is not a power of 2
```

בדוגמה זו –  $1024 = 2^{10}$  ולכן הוא חזקה שלמה של 2 לעומת 3 שאינו חזקה שלמה של 2.

**שימו לב:** במשימה זו מוותר להשתמש ב-bitwise (כגון `<<`, `>>`, `&`, `|`, `^`), חיבור וחסור, פעולות לוגיות (כגון `||`, `&&` וכו'), תנאים (`if, else, switch, case`) אסור להשתמש בלולאות (`while, for, do while`), **כפל, חילוק או מודולו**.

**משימה 5: בדיקה כמה ביטים שונים יש בין שני מספרים**

במשימה זו נקלוט מן המשתמש שני מספרים שלמים (חיוביים, שליליים או אפס) יש למצוא כמה ביטים שונים יש בין שני המספרים (בייצוג הבינארי).

**רמז:** שימוש במשימה 4 יעשה את הפתרון בצורה יעילה ביותר (מומלץ לשימוש אך לא חובה).

ניתן להניח כי הקלט שיוכנס הינו שני מספרים שלמים ואין צורך לבדוק את תקינות הקלט.

דוגמה :

Enter two numbers:

3

5

There are 2 different bits

הסבר : אם נסתכל ב-4 ביטים על 3 ו-5 נקבל :

5=0101, 3=0011 אם נשווה אותם במיקומים נקבל שקיימים שני שונים :

0	1	0	1
0	0	1	1
	X	X	

כאשר X מסמן את הביטים השונים שמספרם הוא 2.

שימו לב לאחר הבקשה: Enter two numbers יש ירידת שורה.

**משימה 6: חיבור בין שני מספרים שלמים**

במשימה זו נקלוט מהמשתמש 2 מספרים שלמים, ונחזיר את תוצאת החיבור ביניהם.

**שימו לב:** במשימה זו מוותר להשתמש ב- bitwise (כגון ^, |, &, <, >, <<, >>) פעולות לוגיות (כגון ==, |, &, <, >) תנאים ולולאות, השימוש ב- (+) ו- (-) מותר, אך אסור להשתמש בפעולות המתמטיות חיבור, חיסור, כפל וחילוק.

**מותר :** ++a, a++, --a, a--

**אסור :** a/b, a-(-b), a\*b, a+b

**הדרכה:** חישבו כיצד מתבצע חיבור עשרוני, לאחר מכן כיצד מתבצע החיבור הבינארי.

דוגמה :

Enter two numbers:

2

-8

-6

שימו לב לאחר הבקשה: Enter two numbers יש ירידת שורה.

**משימה 7: כפל שני מספרים שלמים**

במשימה זו נקלוט מהמשתמש 2 מספרים שלמים, ונחזיר את תוצאת הכפל ביניהם.

**שימו לב:** במשימה זו מותר להשתמש ב-bitwise (כגון ^, |, &, <<, >>), פעולות לוגיות (כגון |, &, ~) וכו' תנאים ולולאות ובפעולה המתמטית חיבור (+), השימוש ב-(++) ו- (--) מותר אך אסור להשתמש בפעולות המתמטיות חיסור, כפל חילוק ומודולו.

דוגמה :

Enter two numbers:

4

-20

-80

שימו לב לאחר הבקשה Enter two numbers: יש ירידת שורה.

# בהצלחה!

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**1**

Enter the sides length:

**0**

|+|

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**1**

Enter the sides length:

**3**

```

/*\
/* \
/* \
| + |
\ * /
\ * /
\ * /

```

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**1**

Enter the sides length:

**5**

```

/*\
/* \
/* \
/* \
/* \
| + |
\ * /
\ * /
\ * /
\ * /
\ * /

```

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**0**

**התוכנית נעצרה**

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**2**

Enter a reversed number in base 20:

**ALT**

Error! L is not a valid digit in base 20

Error! T is not a valid digit in base 20

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**2**

Enter a reversed number in base 20:

**A1J**

7630

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**2**

Enter a reversed number in base 20:

**0**

0

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**3**

Enter a base (2-10):

**2**

Enter a reversed number in base 2:

**10101**

21

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**2**

Enter a reversed number in base 20:

**8**

8

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**734**

Wrong option!

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**3**

Enter a base (2-10):

**8**

Enter a reversed number in base 8:

**76**

55

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**4**

Enter a number:

**2048**

2048 is a power of 2



Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**4**

Enter a number:

**33**

33 is not a power of 2

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**5**

Enter two numbers:

**8**

**10**

There are 1 different bits

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**6**

Enter two numbers:

**0**

**5**

5

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**6**

Enter two numbers:

**-100**

**900**

800

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**7**

Enter two numbers:

**9**

**-9**

-81

Choose an option:

1. Rhombus
2. Base 20 to Decimal
3. Base to Decimal
4. Pow2
5. Different bits
6. Add
7. Multiply
0. Exit

**7**

Enter two numbers:

**-3**

**-7**

21