



הפקולטה למדעי ההנדסה- המחלקה להנדסת תעשייה וניהול
אוטומציה וייצור ממוחשב 364-1-3321
אוניברסיטת בן גוריון בנגב
תרגיל בית 1 Git :

קבוצה: 8

מגשים:

208495226

209193440

316359520

313278418

• **מטרת הפרויקט:**

התבקשנו ליישם את המרה בין בסיס בינארי (2) לבסיס הקסדצימלי (16) ומבסיס הקסדצימלי לבסיס בינארי על ידי פיתוח אלגוריתם משלנו להמרת מספרים מבסיס אחד למשנהו, מבלי להשתמש בפונקציות מובנות. בחרנו לפתח את הקוד ב-JAVA באמצעות תוכנת Eclipse. תחילה המערכת תקבל כקלט מהמשתמש את המספר שברצונו להמיר, תבדוק את תקינותו, תבצע את ההמרה הנבחרת, ולבסוף תציג כפלט למשתמש את המספר בבסיס שנבחר. התוכנית תאכוף שהמשתמש יבחר באפשרות תקינה מתוך התפריט המוצג לו ושהוא מכניס כקלט מספר התואם את האפשרות הנבחרת – מספר שיהיה בהכרח שייך לבסיס המומר ובעל תווים תקינים בלבד.

• **הנחות היסוד בפיתוח המערכת:**

- הנחנו כי הקלט שמוכנס על ידי המשתמש הינו מספר חיובי בלבד ולא יוכנס מספר שלילי.
- הנחנו כי הקלט יהיה בהתאם לנדרש בבסיס (מספרים **שלמים** ותיקנים, ללא סימנים ורווחים).
- לאחר קבלת פלט המשתמש ישאל האם הוא מעוניין לעשות המרת בסיסים נוספת ועל כן הוא יהיה חייב להשיב ערך.
- בהמרה **מהקסדצימלי לבינארי** הקלט היודפס למשתמש יהיה בקבוצות של 4 ספרות כלומר- אם לספרה 1 השמאלית ביותר יהיה חסר אפסים להשלמה לקבוצה של 4 הקוד ישלים את כמות האפסים הנדרשת משמאל לספרה 1.



הפקולטה למדעי ההנדסה- המחלקה להנדסת תעשייה וניהול
אוטומציה וייצור ממוחשב 364-1-3321
אוניברסיטת בן גוריון בנגב

• **תיאור מצבי הקיצון ושיטת הפתרון:**

1. הכנסת אותיות שאינן קיימות בבסיס 16: הקוד ידע לבדוק באמצעות טבלת ASCII אם הקלט שנבחר מכיל אותיות שהם רק בתווח A-F.
2. הכנסת אותיות קטנות – הקוד יודע לבדוק להפוך אות קטנה לגדולה באמצעות השיטה ToUpperCase() ולאחר מכן להחזיר פלט תקין.
3. הזנת ערך לא חוקי או לחיצה על ENTERD ללא ערך מתוך מהתפריט הראשי (בחירת סוג המרת בסיס), תיאכף ותוצא הודעת שגיאה למשתמש עד אשר יזין ערך חוקי.
4. הזנת ערך לא חוקי כאשר המשתמש נשאל אם ברצונו להמיר ערך נוסף תיאכף ותוצא הודעת שגיאה עד אשר יזין בחירת חוקית.

• **תיעוד קוד התוכנית:**

פונקציה decimalToHex()

פונקציה זו מקבלת כקלט מספר בבסיס עשרוני וממירה למספר הקסדצימלי שלו.
עבור הספרות 0 עד 9, היא מחזירה את התו המתאים למספר באמצעות הוספת המספר ל-ASCII של תו '0', מה שייתן את הספרה עצמה. עבור הספרות 10 עד 15, היא מחזירה את התו המתאים על ידי חיבור 10 מהמספר (כיון שאותיות מתחילות מהספרה העשירית), ואז הוספת תו 'A'.

```
public static char decimalToHex(int decimal) {
    // Convert decimal to hexadecimal
    if (decimal >= 0 && decimal <= 9) {
        return (char) (decimal + '0'); //casting from int to char
    } else {
        return (char) (decimal - 10 + 'A'); // casting digits that more than 9 from int to char.
    }
}
```

פונקציה binaryToDecimal()

הפונקציה מקבלת מספר בינארי תקין וממירה אותו למספר בבסיס עשרוני. היא עוברת על ספרות המספר הבינארי מהסוף להתחלה (מימין לשמאל), ומכניסה את כל ספרות המספר הבינארי לחישוב העשרוני.
לכל ספרת בינארי (0 או 1), הפונקציה מבצעת את החישוב הבא:

$$\text{Decimal} = \text{digit}(n) * 2^j + \text{digit}(n-1) * 2^{j+1} + \dots$$



הפקולטה למדעי ההנדסה- המחלקה להנדסת תעשייה וניהול
אוטומציה וייצור ממוחשב 364-1-3321
אוניברסיטת בן גוריון בנגב

כאשר:

- bit הוא ערך הספרה הבינארית הנוכחית (0 או 1).
 - j הוא המיקום של הספרה במספר הבינארי, מתחיל מימין.
- כל ספרת בינארי משמשת כגורם מכפיל ב-(2 בחזקת המיקום שלה), והתוצאה היא סכום של כל המכפולות הללו.

```
public static int binaryToDecimal(String binary) {  
    // Convert binary to decimal  
    int decimal = 0;  
    for (int i = binary.length() - 1, j = 0; i >= 0; i--, j++) { //run on the binary number  
        int bit = binary.charAt(i) - '0'; // convert char to int in ASCII  
        decimal += bit * Math.pow(2, j); //<---looks the digits from the RIGHT (digit1)*2^(j)+(digit2)*2^(j+1)....  
    }  
    return decimal;  
}
```

הפונקציה putZeroBinary() -

הפונקציה מקבלת מחרוזת בינארית ומוודאה שהאורך שלה הוא כפולה של במקרה בו אורך המחרוזת אינו כפולה של 4, היא מוסיפה אפסים לתחילת המחרוזת כך שהאורך יהיה כפולה של 4. כמו כן, על מנת לדעת כמה ספרות יש להוסיף נעשה מודלו לאורך מחרוזת והשארית שתתקבל היא כמות הספרות שנדרש להוסיף. במקרה בו השארית אינה 0 נוסיף את כמות הספרות הנדרשת בתחילת מחרוזת חדשה (FitedBinary) על ידי לולאה כלומר בכל איטרציה של הלולאה, נוסיף את הספרה '0' לתחילת המחרוזת. לאחר איטרציות אלו יתווסף המחרוזת המקורית לקצה הימני של מחרוזת האפסים וזו התוצאה שתתקבל.

```
public static String putZeroBinary(String binaryNumber) {  
    // Ensure the binary number has a length that is a multiple of 4  
    int extraBinaryDigits = binaryNumber.length() % 4; // Represents the extra binary digits when dividing binaryNumber length by 4  
    if (extraBinaryDigits != 0) {  
        int DigitsNeedToAdd = 4 - extraBinaryDigits;  
        String FitedBinary = ""; // Initialize as an empty string  
        for (int i = 0; i < DigitsNeedToAdd; i++) {  
            FitedBinary += '0'; // Concatenate '0' to the string  
        }  
        FitedBinary += binaryNumber; // Concatenate binaryNumber to the string  
        binaryNumber = FitedBinary; // Assign the concatenated string back to binaryNumber  
    }  
    return binaryNumber;  
}
```

הפונקציה isValidHex() - פונקציה זו בודקת האם המספר שהוזן על ידי המשתמש הינו בבסיס הקסדצימלי. בהינתן קלט מהמשתמש תתבצע בדיקה עבור כל תו בקלט תתבצע בדיקה שכל תו במחרוזת הוא ספרה בטווח 0-9 או A-F (או a-f) ושהמחרוזת אינה ריקה. אם התנאים מתקיימים, הפונקציה מחזירה true, אחרת היא מחזירה false.



הפקולטה למדעי ההנדסה- המחלקה להנדסת תעשייה וניהול
אוטומציה וייצור ממוחשב 364-1-3321
אוניברסיטת בן גוריון בנגב

```
static boolean isValidHex(String hexNumber) {  
    if (hexNumber=="") {  
        return false;  
    }  
    // Check if the input contains valid hexadecimal char ( 0-9 and A-F)  
    for (char c : hexNumber.toCharArray()) {  
        if (!Character.isDigit(c) && (Character.toUpperCase(c) < 'A' || Character.toUpperCase(c) > 'F')) {  
            return false;  
        }  
    }  
    return true;  
}
```

הפונקציה isValidBinary()

פונקציה זו בודקת האם המספר שהוזן על ידי המשתמש הינו בבסיס בינארי. בהינתן קלט מהמשתמש ראשית תתבצע בדיקה שהקלט אינו ריק ולאחר מכן תתבצע בדיקה נוספת עבור כל תו בקלט תתבצע בדיקה תו הוא 0 ו-1 בלבד. במידה והקלט ריק או נמצא תו שאינו תקין הפונקציה מחזירה False בהתאם אחרת תחזיר True.

```
public static boolean isValidBinary(String binaryNumber) {  
    if (binaryNumber=="") {  
        return false;  
    }  
    // Check if the input contains only 0s and 1s  
    for (char c : binaryNumber.toCharArray()) {  
        if (c != '0' && c != '1') {  
            return false;  
        }  
    }  
    return true;  
}
```

הפונקציה convertDecimalToBinary()

פונקציה זו מקבלת כקלט מספר בבסיס דצימאלי וממירה אותו למספר בינארי. תחילה, היא מבצעת את ההמרה על ידי חילוק חוזר של המספר העשרוני ב-2 ושמירת השארית בכל שלב. השאריות משורשרות במחרוזת מהסוף להתחלה (מהשארית האחרונה לראשונה), והלולאה ממשיכה עד שהמספר העשרוני הופך ל-0 (לא נותר מה לחלק). בסיום הלולאה, הפונקציה מחזירה את המחרוזת הבינארית המייצגת את המספר העשרוני.



הפקולטה למדעי ההנדסה- המחלקה להנדסת תעשייה וניהול
אוטומציה וייצור ממוחשב 364-1-3321
אוניברסיטת בן גוריון בנגב

```
public static String convertDecimalToBinary(int decimal) {  
    // Convert decimal to binary  
    if (decimal == 0) {  
        return "0";  
    }  
  
    String binary = "";  
    while (decimal > 0) {  
        int remainder = decimal % 2;  
        binary = remainder + binary; //take all the remainders. add the current remainder to the left every time  
        decimal /= 2;  
    }  
    return binary;  
}
```

הפונקציה convertHexToBinary()

פונקציה זו מקבלת כקלט מספר בבסיס הקסדצימלי וממירה אותו למספר בינארי. תחילה, היא עוברת על כל תו במחרוזת ההקסדצימלית באמצעות לולאה, ממירה אותו למספר עשרוני ולאחר מכן ממירה את המספר העשרוני למספר בינארי באמצעות פונקציה נוספת ששמה `convertDecimalToBinary()`. התוצאות מווספות יחד על ידי שרשרת שמייצג מחרוזת תוך הקפדה על התווספות אותיות גדולות (ע"י `ToUpperCase()`) וספרות לתוצאה סופית (מחרוזת) המייצגת ספרות בינאריות בלבד. לפני הוספת המספר הבינארי לתוצאה, נוודא שהוא מורכב מכמות ספרות שהן כפולה של 4 באמצעות פונקציה `putZeroBinary`.

```
public static String convertHexToBinary(String hexNumber) {  
    // Convert each hex digit to binary  
    String binaryNumber = "";  
    for (int i = 0; i < hexNumber.length(); i++) {  
        // Get the current hex digit  
        char hexDigit = hexNumber.charAt(i);  
        // Calculate decimal value based on a digit or a letter  
        int decimalValue;  
        if (Character.isDigit(hexDigit)) {  
            // If it's a digit, convert directly to decimal  
            decimalValue = hexDigit - '0'; // ASCII  
        } else {  
            // If it's a letter (A-F), convert after making sure it is an uppercase letter  
            decimalValue = Character.toUpperCase(hexDigit) - 'A' + 10; // and then convert from hexadecimal to decimal  
        }  
        // Convert decimal value to binary  
        String binaryValue = convertDecimalToBinary(decimalValue);  
        binaryValue = putZeroBinary(binaryValue);  
        // Add the binary value to the string result  
        binaryNumber += binaryValue;  
    }  
    return binaryNumber;  
}
```

הפונקציה convertBinaryToHex()

פונקציה זו ממירה מספר בינארי למספר הקסדצימלי. היא עוברת על כל 4 ספרות בינארי (תת-מחרוזת) במספר הבינארי, ממירה את התת-מחרוזת למספר עשרוני, ולאחר מכן ממירה את המספר העשרוני למספר הקסדצימלי. גם כאן יש שימוש בשיטה `putZeroBinary`, שמבצעת הוספת אפסים כדי לוודא שאורך המספר הבינארי הוא בכפולות של 4.

המרת תת-מחרוזת למספר עשרוני:



הפקולטה למדעי ההנדסה- המחלקה להנדסת תעשייה וניהול

אוטומציה וייצור ממוחשב 364-1-3321

אוניברסיטת בן גוריון בנגב

מציין תת-מחרוזת של 4 תווים בינארי String bin = binaryNumber.substring(i, i + 4)

int decimalValue = binaryToDecimal(bin)

מקבל את התת-מחרוזת בינארית וממיר אותה למספר עשרוני באמצעות פונקציה binaryToDecimal.

המרת מספר עשרוני להקסדצימלי:

hexadecimalNumber += decimalToHex(decimalValue): ממירה את המספר העשרוני להקסדצימלי

באמצעות פונקציה decimalToHex ומוסיפה אותו לתוצאה.

החזרת תוצאה סופית:

return hexadecimalNumber.toString הפונקציה מחזירה את התוצאה הסופית- המספר ההקסדצימלי המלא.

```
public static String convertBinaryToHex(String binaryNumber) {

    // make sure the length of the binary number is a multiple of 4 by converting to X4 digits
    binaryNumber = putZeroBinary(binaryNumber);

    // Convert binary to hexadecimal
    String hexadecimalNumber = ""; // Change the binary number to string number
    for (int i = 0; i < binaryNumber.length(); i += 4) {
        String bin = binaryNumber.substring(i, i + 4); // For every 4 binary digits - convert to decimal
        int decimalValue = binaryToDecimal(bin);
        hexadecimalNumber += decimalToHex(decimalValue); // Add all converts of all 4 digits groups to one string
    }

    return hexadecimalNumber.toString();
}
```

• תיעוד פלט תוכנית:

תחילה יוצג תפריט למשתמש בו יבחר את הבסיסים מהם ירצה להמיר:

```
Choose number base to convert:
1. Binary to Hexadecimal
2. Hexadecimal to Binary
0. Exit
Enter your choice (0, 1, or 2):
```

במידה והוזן קלט שלא מהווה את אחת מאופציות התפריט תוצג לו הודעת שגיאה והוראה לבחור

שנית מתוך התפריט:

```
Enter your choice (0, 1, or 2): 5
Invalid choice. Please enter 0, 1, or 2.
```



הפקולטה למדעי ההנדסה- המחלקה להנדסת תעשייה וניהול
אוטומציה וייצור ממוחשב 364-1-3321
אוניברסיטת בן גוריון בנגב

לאחר בחירה באפשרות תקינה, יוצג שוב התפריט:

```
1
Choose number base to convert:
1. Binary to Hexadecimal
2. Hexadecimal to Binary
0. Exit
Enter your choice (0, 1, or 2):
```

כאשר נבחר באפשרות 1, המחשבון יבקש מספר בינארי להמרה, וימיר אותו להקסדצימלי.
כאשר נבחר באפשרות 2, המחשבון יבקש מספר הקסדצימלי להמרה, וימיר אותו לבינארי.
בבחירת מספר בינארי/הקסדצימלי **לא תקין** בהתאמה, תצא הודעת שגיאה והמשתמש יתבקש להזין מספר נוסף

```
Choose number base to convert:
1. Binary to Hexadecimal
2. Hexadecimal to Binary
0. Exit
Enter your choice (0, 1, or 2): 2
Enter a hexadecimal number: q
Invalid hexadecimal input
Enter a hexadecimal number:
```

```
Choose number base to convert:
1. Binary to Hexadecimal
2. Hexadecimal to Binary
0. Exit
Enter your choice (0, 1, or 2): 1
Enter a binary number: 29
Invalid binary input
Enter a binary number:
```

בעבור הזנת קלט הקסדצמאלי תקין, המחשבון יבצע את ההמרה וישאל אם לבצע המרה נוספת:

```
Enter a hexadecimal number:
4f
Binary: 01001111
Do you want to convert another number? (1 for yes/ 0 for no):
```

ובעבור הזנת קלט בינארי תקין, המחשבון יבצע את ההמרה וישאל אם לבצע המרה נוספת:

```
Enter a binary number: 10101
Hexadecimal: 15
Do you want to convert another number? (1 for yes/ 0 for no):
```

בעבור לחיצה על 1, יוצג למשתמש התפריט הראשי שוב:

```
Do you want to convert another number? (1 for yes/ 0 for no): 1
Choose number base to convert:
1. Binary to Hexadecimal
2. Hexadecimal to Binary
0. Exit
Enter your choice (0, 1, or 2):
```



הפקולטה למדעי ההנדסה- המחלקה להנדסת תעשייה וניהול
אוטומציה וייצור ממוחשב 364-1-3321
אוניברסיטת בן גוריון בנגב

כאשר, בעבור לחיצה על 0, התוכנית תסתיים:

```
Enter your choice (0, 1, or 2): 0  
you chose to exit, Goodbye <3
```

במענה על השאלה " האם תרצה להמיר מספר נוסף", אם תשובת המשתמש אינה תקינה (מספר שאינו 0 או 1)-
יוצג הפלט הבא, שיציג הודעת שגיאה ויבקש בחירה תקינה.

```
Do you want to convert another number? (1 for yes/ 0 for no): 9  
Enter a valid choice 1
```

לאחר לחיצה על 1 (להמשיך להזנת פלט נוסף), יוצג שוב התפריט הראשי, ועל המשתמש לבחור את סוג ההמרה
הרצוי:

```
Choose number base to convert:  
1. Binary to Hexadecimal  
2. Hexadecimal to Binary  
0. Exit  
Enter your choice (0, 1, or 2):
```

בלחיצה על 0- תבוצע יציאה מהתוכנית:

```
Do you want to convert another number? (1 for yes/ 0 for no): 0  
you chose to exit, Goodbye <3
```




הפקולטה למדעי ההנדסה- המחלקה להנדסת תעשייה וניהול
אוטומציה וייצור ממוחשב 364-1-3321
אוניברסיטת בן גוריון בנגב

• **הוראות למפעיל נספח 1:**

1. הרץ את התוכנית

2. בחר את האפשרות הרצויה מהתפריט ולחץ ENTER

1. המרת מספר מבסיס דצימלי הקסדצימלי

2. המרת מספר מבסיס הקסדצימלי לדצימלי

0. יציאה

```
Choose number base to convert:
1. Binary to Hexadecimal
2. Hexadecimal to Binary
0. Exit
Enter your choice (0, 1, or 2):
```

3. אם בחרת 1 או 2, הזן את המספר אותו תרצה להמיר ולחץ ENTER . אם בחרת 0, התוכנית תיסגר.

בהזנת קלט לא תקין, תתבקש להזין שוב.

4. המספר המומר יוצג על המסך

```
Enter a hexadecimal number: A12
Binary: 101000010010
```

5. במידה והמשתמש מעוניין להמיר מספר נוסף: יש להזין את הספרה 1 וללחוץ ENTER . התפריט הראשי יוצג מחדש.

```
Do you want to convert another number? (1 for yes/ 0 for no):
```

6. ליציאה: הקש 0. עבור הזנה לא תקינה- תתבקש להזין בחירה תקנית (0 או 1).