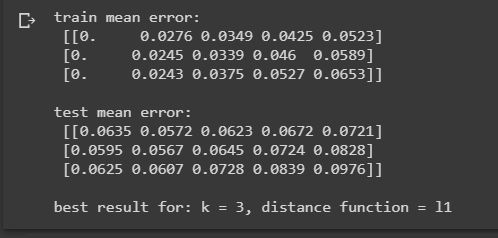
# למידת מכונה - מטלה 4

מאיר נזרי - 312237563

## K-Nearest Neighbor

השתמשתי בגרסה Python 3.8.

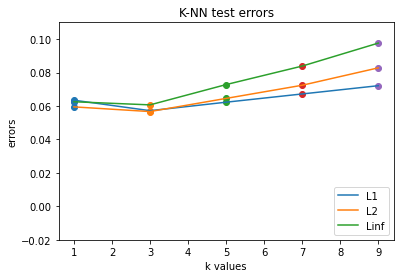
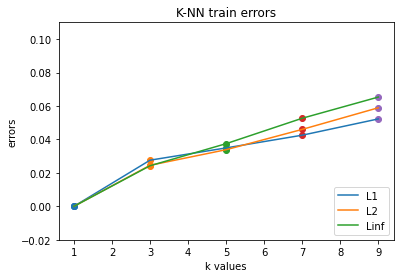
לאחר הרצת הקוד אלו התוצאות. כל שורה מייצגת תוצאות של פונקציית מרחק נפרדת. סדר השורות מלמעלה למטה הם . כל עמודה מייצגת תוצאות של ערך k שונה בסדר עולה משמאל לימין. פירוט נוסף של התוצאות בגרפים למטה.

**מהם הפרמטרים p, k הטובים ביותר?**

כמו שניתן לראות ערכי ההיפר-פרמטרים הטובים ביותר על ה-test הם עבור k=3 ו-p=2, עם טעות מינימלית של 5.67%.

**האם יש Overfitting?**

אכן יש כאן Overfitting. בגרפים למטה ניתן לראות שיש פער ניכר בין רמות הדיוק ב-train וב-test עבור ערכי k נמוכים. אולם ככל שהערך של k גדל פער זה הולך מצטמצם, ואז יש יותר חשש ל-underfitting.



קישור ל-Colab של מחברת הקוד:

<https://colab.research.google.com/drive/19fAZecWymU22ciBKIyNpUL1GsFPfMs-m#scrollTo=YsVXNY2bd96D>

## Huffman encoding

1. לאחר הרצת אלגוריתם האפמן על התדירויות התקבל הקוד למטה. העץ האפמן שנוצר מוצג בתמונה בעמוד הבא.

א – 1010

ב – 11111

ג – 000101

ד – 00011

ה – 010

ו – 001

ז – 100100

ח – 111101

ט – 000100

י – 011

כ – 10111

ל – 1100

מ – 1101

נ – 11100

ס – 101100

ע – 10011

פ – 101101

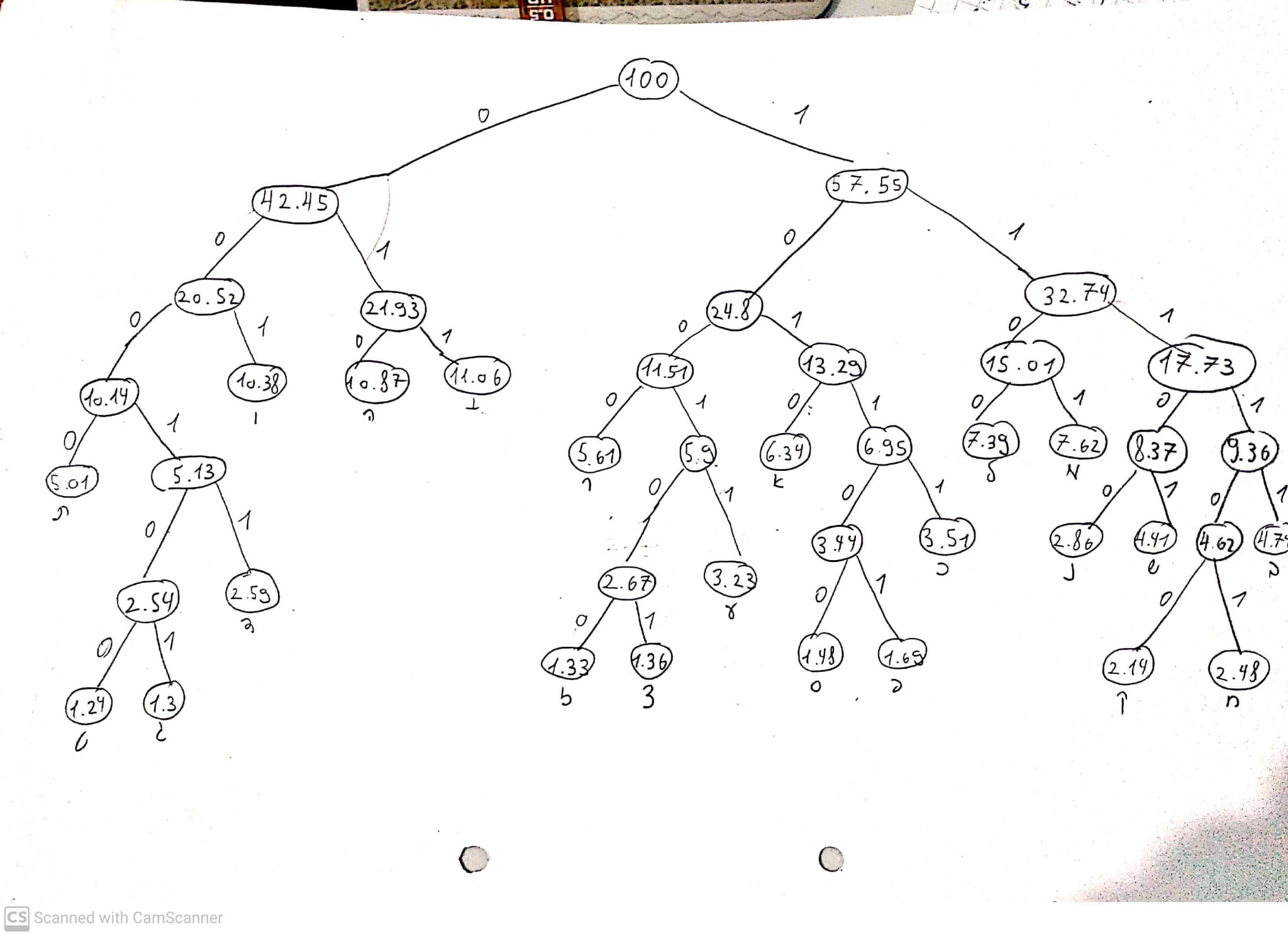
צ – 100101

ק – 111100

ר – 1000

ש – 1101

ת – 0000



1. קידוד השם "מאיר" לפי הקוד לעיל הוא: "1101 1010 011 1000".
2. התדירות של כל אחת מארבעת האותיות בשם "מאיר" היא בדיוק 0.25% ולכן חישוב האנתרופיה יהיה כך:

שאנון הוכיח שלא יכולה להיות דחיסה שאורך התו הממוצע שלה יותר טוב מהאנתרופיה. ולכן הדחיסה האופטימלית לשם "מאיר" היא לכל הפחות באורך של 8 ביטים. לעומת זאת, בסעיף 2 למרות שהשתמשנו באלגוריתם האפמן שהוא אופטימלי, אורך הקידוד הוא 15 ביטים. הסיבה להפרש היא שבסעיף 2 התחשבנו ב22 תווים עם תדירויות הרבה יותר קטנות מאשר איך שחישבנו את האנתרופיה כאן. כתוצאה מכך, האי-וודאות (רנדומליות) הרבה יותר גדול ולכן האנתרופיה גדולה הרבה יותר.