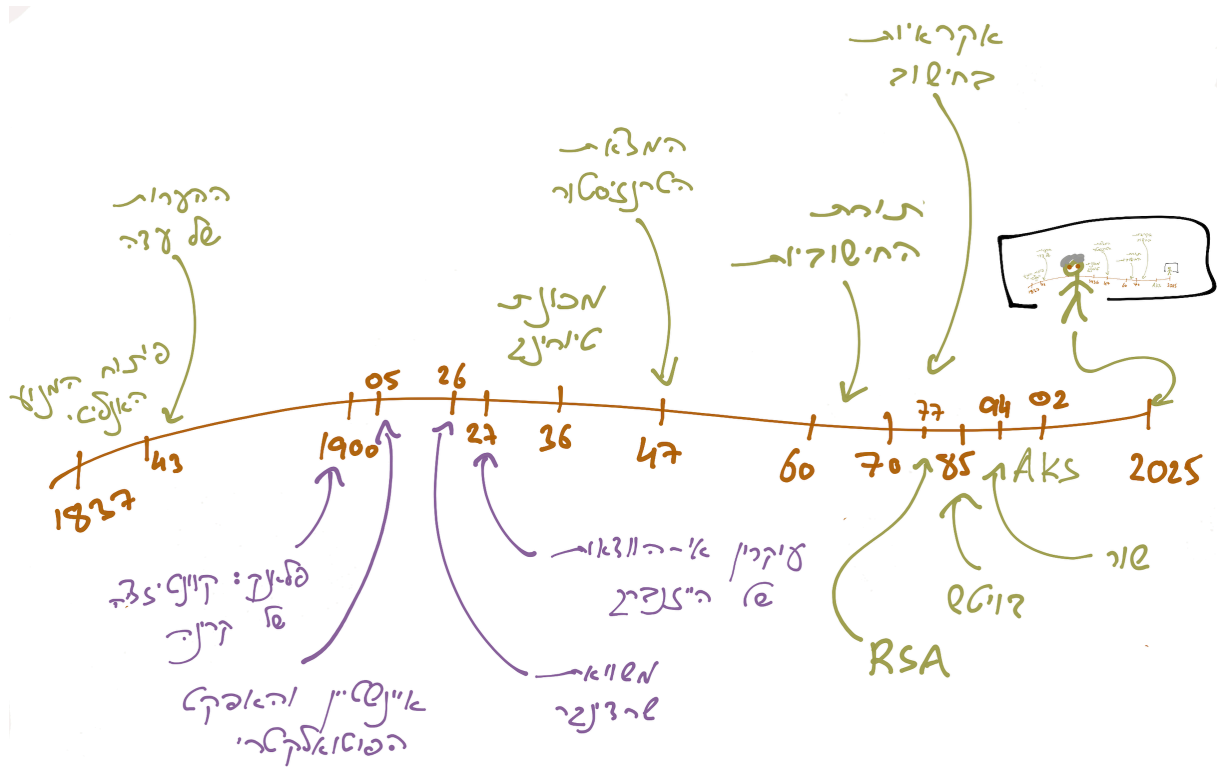


ברוכים הבאים לקורס אלגוריתמים קוונטיים!

## 1 חישוב בראי היסטורי

לפני שנדון בחישוב קוונטי, אבקש לקחת צעד אחורה ולעסוק במושג החישוב עצמו. כך נוכל למקם את החישוב הקוונטי בהקשר הראוי, הן מבחינה רעיונית והן מבחינה היסטורית.



### 1.1 טיורינג וצ'רץ'

במובן מסוים, מאמרו המפורסם של אלן טיורינג משנת 1936, On Computable Numbers, מסמן את רגע הלידה של תורת החישוביות. במאמר זה הציע טיורינג מודל מתמטי פשוט להפליא, שנועד ללכוד את מושג החישוב באמצעות הליך מכני. הרעיון הוא שכל חישוב שאפשר לבצע לפי הוראות חד-משמעיות, צעד אחר צעד,

אפשר לבצע גם באמצעות מכונת טיורינג. זוהי, בניסוח לא פורמלי, תזת צ'רץ'-טיורינג (Church-Turing thesis). זו אינה טענה על מהירות החישוב, אלא על עצם האפשרות לחשב.

אחת התובנות המפתיעות במאמר היא שאין צורך לבנות מכונה אחרת לכל משימה. אפשר לבנות מכונה אוניברסלית, המקבלת תיאור של מכונה אחרת ואת הקלט שלה, ומדמה את פעולתה. במונחים של היום: המכונה קבועה, ומה שמשתנה הוא התוכנית.

טיורינג, עם זאת, לא פעל בחלל ריק. אלונזו צ'רץ' הציע באותה תקופה פורמליזציה אחרת של מושג החישוב, המבוססת על תחשיב למבדא. עבודתו פורסמה סמוך להגשת מאמרו של טיורינג, שפיתח את רעיונותיו באופן בלתי תלוי. טיורינג הראה ששני המודלים שקולים בכוחם החישובי. השקילות הזו היא משפט מתמטי; הזיהוי של המודלים עם המושג האינטואיטיבי של חישוב הוא התזה. בהמשך היה צ'רץ' המנחה של טיורינג לדוקטורט בפרינסטון, שאותו השלים בשנת 1938. כמו היום, גם אז פרינסטון היה מקום מכובד למדי לעשות בו מתמטיקה; באותן שנים פעלו בו ובמכון למחקר מתקדם הסמוך חוקרים כמו קורט גדל וג'ון פון נוימן.

## 1.2 עדה ובג'

אבל אם כבר מסתכלים אחורה, כדאי ללכת עוד קצת. כמעט מאה שנה קודם לכן פעלו צ'ארלס בבג' (Charles Babbage) ועדה לאבלייס (Ada Lovelace). בשנת 1843 פרסמה לאבלייס תרגום של מאמר מאת לואיג'י מנבראה על המנוע האנליטי של בבג', והוסיפה לו "הערות" מקיפות, ארוכות מן המאמר עצמו. בהערות הסבירה את עקרונות הפעולה של המכונה וצירפה תוכנית מפורטת לחישוב מספרי ברנולי. זו נחשבת לאחת מאבני הדרך הראשונות בתולדות התכנות, ולעתים מתוארת כתוכנית המחשב הראשונה שפורסמה.

מה שמרשים במיוחד הוא שלאבלייס ראתה מעבר לחישובים במספרים. היא הבינה שמכונה כזו יכולה לפעול על סימנים לפי כללים, ולכן, אם נדע לייצג באמצעותם יחסים מתאימים, היא תוכל לעסוק גם בדברים אחרים. הדוגמה שנתנה הייתה מוזיקה: עקרונית, המכונה תוכל לחבר יצירות אם ננסח את היחסים המוזיקליים באופן מתאים. היא הבחינה בין הפעולות לבין העצמים שעליהם הן פועלות, ודיברה על "מדע הפעולות", science of operations. קשה שלא להתרשם מן החזון הזה, בתקופה שבה המכונה עצמה עדיין הייתה בעיקר על הנייר.

לשם קנה מידה היסטורי: כל זה התרחש כ-90 שנה לפני עבודתו של טיורינג, בערך אותו פרק זמן שמפריד בין טיורינג לבינינג. טיורינג, אגב, חזר לרעיונותיה של לאבלייס במאמרו משנת 1950, Computing Machinery and Intelligence, במסגרת הדיון בשאלה אם מכונות יכולות לחשוב. בפרט, הוא דן בטענה שמכונה יכולה לעשות רק מה שאנחנו יודעים להורות לה לעשות, ובקשר של טענה זו ליצירתיות וללמידה.

בבג' היה מתמטיקאי וממציא בריטי, שלא הסתפק בחזון וניסה לממש אותו במכונה ממשית. בשנת 1822 החל בפרויקט של מנוע ההפרשים, מכונה שנועדה לחשב ולהפיק טבלאות מתמטיות באמצעות שיטת ההפרשים הסופיים. הפרויקט נתקל בקשיים הנדסיים, כספיים וארגוניים, ובניית המכונה המלאה לא הושלמה. מעניין לציין כי בשנת 1991 השלים מוזיאון המדע בלונדון את בניית מנגנון החישוב של מנוע ההפרשים מספר 2, לפי תכנון מאוחר יותר של בבג' מן השנים 1847-1849.

בשנת 1834 כבר הגה בבג' רעיון שאפתני עוד יותר: **המנוע האנליטי**. בשנת 1837 היה בידי תכנון מפורט, והוא המשיך לפתח את הרעיון עד מותו ב-1871. למכונה הזו כבר נועדו להיות רבות מן התכונות שאנחנו מזהים עם מחשב כללי: יחידת עיבוד, זיכרון, קלט ופלט, תוכנית המוזנת בכרטיסים מנוקבים, ואפילו קפיצות מותנות ולולאות. באחד מתכנוניו נועד הזיכרון להכיל אלף מספרים בני חמישים ספרות עשרוניות כל אחד. זהו יעד הנדסי שאפתני למכונה מכנית, בוודאי באמצע המאה ה-19.

הדמיון למחשב מודרני מרשים, אבל אין להסיק ממנו קו התפתחות הנדסי רציף. המנוע האנליטי לא הושלם, ורבים מרעיונותיו הופיעו מחדש במחשבים של המאה ה-20. ההערות של לאבלייס היו במשך שנים מקור מרכזי להבנת הרעיון; התוכניות והרשימות הרבות שהשאיר בבג' חשפו בהמשך עד כמה רחוק הגיע בתכנון.

### 1.3 בין רעיון למכונה: חישוב מופשט ומחשבים ממשיים

הקשר בין התאוריה למעשה, בין הפשטת מושג החישוב לבין האתגרים הכרוכים בבניית מכונה שמממשת אותה, הוא מוטיב חוזר בחקר המחשב. הוא רלוונטי במיוחד לחישוב קוונטי: אנחנו יודעים לתאר אלגוריתמים מעניינים, אבל היכולת להריץ אותם בקנה המידה הרצוי מחייבת להתגבר על קשיים הנדסיים עצומים. ההתקדמות בצד ההנדסי מעוררת, מטבע הדברים, התלהבות גם אצל התאורטיקנים.

אם נחזור לרגע למחשוב הקלאסי, אחת מפריצות הדרך ההנדסיות החשובות התרחשה בשנת 1947: ג'ון ברדין וולטר ברטיין בנו במעבדות בל את הטרנזיסטור הראשון, במסגרת קבוצת המחקר שהוביל ויליאם שוקלי. שוקלי פיתח בהמשך את טרנזיסטור הצומת, ושלושתם חלקו את פרס נובל לפיזיקה בשנת 1956. הטרנזיסטור פתח את הדרך למחשבים קטנים, אמינים וחסכוניים בהרבה.

בקורס הזה נתמקד בצד המופשט של חישוב קוונטי. לא נעסוק בהיבטים ההנדסיים המרתקים של בניית מחשב כזה, אבל כדאי לזכור שהמתמטיקה שאנחנו מפתחים אמורה, בסופו של דבר, לתאר משהו שהטבע מאפשר לעשות.

### 1.4 יעילות

עד כה שאלנו מה אפשר לחשב. זו שאלה יסודית, אבל מבחינה מעשית היא אינה מספיקה: אלגוריתם שמסתיים אחרי יותר שנים מגיל היקום אינו מועיל לנו במיוחד. בשנות ה-60 וה-70, בעבודות של קובהאם, אדמונדס, הרטמניס, סטרנס, בלום ואחרים, התבססה תורת הסיבוכיות, העוסקת במשאבים הדרושים לביצוע חישוב. בפרט, השאלה "מה ניתן לחשב?" פינתה מקום גם לשאלה "מה ניתן לחשב ביעילות?".

כמקובל בתורת הסיבוכיות, נשתמש בזמן ריצה פולינומי באורך הקלט כאמת מידה ליעילות. זו כמובן הפשטה: לא כל פולינום מבטיח אלגוריתם שימושי, אבל ההבחנה בין זמן פולינומי לזמן גדול מפולינומי התגלתה כפורייה ויציבה מאוד. מחלקת בעיות ההכרעה הניתנות לפתרון דטרמיניסטי בזמן פולינומי מסומנת  $P$ .

בהקשר זה נוסחה גם התזה החזקה של צ'רץ'-טיורינג. בגרסה דטרמיניסטית שלה, כל מודל חישוב "סביר" ניתן לסימולציה במכונת טיורינג בתוספת פולינומית בלבד לזמן הריצה. למילה "סביר" יש כאן תפקיד חשוב: אנחנו מבקשים לתאר חישוב שאפשר לממש, לפחות עקרונית, ולא מודל שמקבל בחינם פעולות בעלות כוח בלתי מוגבל. בהמשך נפגוש גם את הגרסה ההסתברותית של התזה.

שימו לב להבדל: תזת צ'רץ'-טיורינג המקורית אינה דורשת שהסימולציה תהיה יעילה. התזה החזקה מוסיפה בדיוק את הדרישה הזו, והיא שהופכת את הסיפור למעניין במיוחד בהקשר הקוונטי.

#### 1.4.1 מחשוב אנלוגי

אתגר אחד מגיע ממחשוב אנלוגי, שבו מתארים את המידע באמצעות גדלים רציפים. במודלים מתמטיים מסוימים מסוג זה אפשר לפתור בעיות מהר מאוד, ובחלקם אפילו לחשב פונקציות שאינן ניתנות לחישוב במכונת טיורינג. אבל לפני שמתרגשים, צריך לבדוק מה המודל קיבל בחינם. האם מותר לשמור מספר ממשי בדיוק אינסופי? האם ניתן לקרוא ספרה עמוקה כרצוננו בזמן קבוע? האם מידע בלתי-חשובי כבר הוחבא בקבועים של המערכת? כשמביאים בחשבון את עלות הדיוק ואת השפעת הרעש, חלק מן היתרונות האלה נעלמים; התוצאה המדויקת תלויה במודל ובהנחות. הלקח אינו שכל מודל אנלוגי חסר עניין, אלא שגם הפשטה מתמטית צריכה לתת דין וחשבון על המשאבים שהיא מניחה ועל המגבלות הפיזיקליות של מימושה.

#### 1.4.2 הטלות מטבע

באמצע שנות ה-70 התברר עד כמה היכולת להטיל מטבע יכולה להיות שימושית לאלגוריתם. דוגמה מפורסמת היא בדיקת ראשוניות. בשנת 1977 פרסמו סולוביי ושרסן מבחן ראשוניות אקראי בזמן פולינומי; באותן שנים פיתח גם רבין מבחן אקראי המבוסס על עבודתו של מילר. באותה תקופה לא היה ידוע אלגוריתם דטרמיניסטי בזמן פולינומי לבעיה ללא הנחות נוספות. לכאורה, משאב פשוט כמו הטלת מטבע מאפשר לנו לעשות משהו שלא ידענו לעשות בלעדיו.

הכוונה בזמן פולינומי היא לזמן פולינומי במספר הספרות של הקלט, ולא בערכו של המספר הנבדק. בדיקה של כל המחלקים האפשריים אינה עומדת בדרישה הזו. במבחנים האקראיים אנחנו מרשים הסתברות קטנה לטעות, ואפשר להקטין אותה כרצוננו באמצעות חזרות בלתי תלויות.

כבר ב-1976 פרסם מילר מבחן דטרמיניסטי בזמן פולינומי בהנחת השערת רימן המורחבת. זו הייתה סיבה טובה לקוות שאפשר יהיה להיפטר מן האקראיות לפחות בבעיה הזו. ואכן, בשנת 2002 הציגו אגרוואל, קאיאל וסקסנה את האלגוריתם המוכר בשם AKS, במאמר PRIMES is in P שפורסם בכתב עת בשנת 2004. זהו אלגוריתם דטרמיניסטי בזמן פולינומי לבדיקת ראשוניות, ללא השערות לא מוכחות. כך נסגר המעגל במקרה חשוב אחד; השאלה הכללית נותרה פתוחה.

מכיוון שאקראיות היא משאב שאפשר לממש באופן פיזיקלי, טבעי לאפשר אותה גם למודל החישוב הבסיסי שלנו. כך מתקבלת הגרסה ההסתברותית של התזה החזקה: כל מודל חישוב סביר ניתן לסימולציה יעילה באמצעות מכונת טיורינג הסתברותית. מחלקת בעיות ההכרעה הניתנות לפתרון בזמן פולינומי באלגוריתם אקראי, עם הסתברות לטעות שלכל היותר  $1/3$  בכל קלט, מסומנת BPP. הקבוע המסוים אינו חשוב; באמצעות חזרות אפשר להקטין את השגיאה.

מרגע שנכנסה למסגרת של תורת הסיבוכיות, האקראיות עצמה הפכה למושא מחקר. בפרט, פותחו מחלצי אקראיות, randomness extractors, המאפשרים להפיק ביטים כמעט אחידים ממקורות אקראיות חלשים, תחת הנחות

מתאימות. למשל, אפשר להיעזר בזרע קצר של ביטים אחידים ועצמאיים, או לנצל עצמאות בין כמה מקורות חלשים. בלי שום הנחה נוספת, אי אפשר פשוט "לזקק" כל מקור חלש יחיד באמצעות אלגוריתם דטרמיניסטי.

במקביל התפתח קשר מפתיע בין קושי חישובי לבין אקראיות. אבן דרך מרכזית היא עבודתם של ניסן וויגדרוזן, Hardness vs Randomness, שהוצגה ב-1988 ופורסמה בכתב עת ב-1994. הרעיון, באופן גס, הוא שפונקציה שקשה לחשב יכולה לשמש לבניית מחולל פסאודו-אקראי: אלגוריתם יעיל לא יצליח להבחין בין הפלט שלו לבין ביטים אקראיים באמת. תחת הנחות מתאימות של קושי חישובי, קו המחקר הזה מאפשר לסמלץ אלגוריתמים אקראיים באופן דטרמיניסטי בזמן פולינומי.

ההשערה המקובלת היא אפוא

$$BPP = P.$$

אבל זו עדיין השערה. הקשרים לחסמים תחתונים למעגלים מספקים סיבות טובות להאמין בה, וגם מסבירים מדוע חלק מן הדרכים הטבעיות להוכיח אותה קשות כל כך. גם אם השוויון נכון, אין פירושו שאקראיות חסרת תועלת: היא עשויה לפשט אלגוריתמים ולקצר מאוד את זמן הריצה שלהם. הטענה היא שבבעיות הכרעה היא אינה מאפשרת מעבר מבעיה שאין לה אלגוריתם פולינומי לבעיה שיש לה כזה.

### 1.4.3 חישוב קוונטי

האם יש משאב פיזיקלי נוסף שהמודל שלנו מפספס? כאן אנחנו מגיעים לחישוב קוונטי. הרעיון פשוט, וכבר נגענו בו: חישוב הוא בסופו של דבר תהליך פיזיקלי. כדי לבצע חישוב, אנחנו משנים באופן מבוקר את מצבה של מערכת ממשית. לכן השאלה אילו חישובים אפשר לבצע, ובאילו משאבים, צריכה להתכתב עם חוקי הפיזיקה.

בתחילת שנות ה-80 החלו לבחון ברצינות מה קורה כשמכניסים לתמונה את המכניקה הקוונטית. פול בניאוף הציג בשנת 1980 תיאור קוונטי של חישוב במכונת טיורינג. ריצ'רד פיינמן הדגיש, בהרצאה מ-1981 ובמאמר מ-1982, את הקושי לסמלץ מערכות קוונטיות באמצעות מחשב קלאסי, והציע להיעזר במערכות קוונטיות כדי לבצע את הסימולציה. בשנת 1985 הציע דייוויד דויטש מודל של מחשב קוונטי אוניברסלי, וניסח במפורש את הקשר בין תזת צ'רץ'-טיורינג לבין העקרונות הפיזיקליים שעליהם נשען החישוב.

התברר שאפשר לנצל תופעות קוונטיות כדי לקבל יתרונות אלגוריתמיים. דויטש, ואחריו חוקרים נוספים, הציגו בעיות שבהן אלגוריתם קוונטי זקוק לפחות שאילתות מאלגוריתם קלאסי. ההשוואה נעשתה במודל האורקל, oracle model, שבו מקבלים גישה לפונקציה כאל קופסה שחורה וסופרים כמה פעמים צריך לפנות אליה. נראה דוגמאות כאלה בקורס. חשוב כבר עכשיו להבחין: הפרדה במודל הזה היא תוצאה מתמטית ממשית, אבל היא אינה מוכיחה כשלעצמה הפרדה בין מחלקות הסיבוכיות הרגילות.

נקודת המפנה הדרמטית הגיעה בשנת 1994, כשפיטר שור הציג אלגוריתמים קוונטיים בזמן פולינומי לפירוק מספרים לגורמים ולחישוב לוגריתם דיסקרטי. אלה כבר בעיות טבעיות ובעלות חשיבות רבה, ולא רק בעיות שהוגדרו כדי להמחיש יתרון במודל האורקל. בפרט, אלגוריתם יעיל לפירוק לגורמים מאפשר לשבור את הצפנת RSA באמצעות שחזור המפתח הפרטי.

כדאי להבחין כאן בין שתי בעיות שונות: לבדוק אם מספר הוא ראשוני, שכבר ראינו שאפשר לעשות בזמן פולינומי במחשב קלאסי, ולמצוא את גורמיו הראשוניים, שלא ידוע לה אלגוריתם קלאסי בזמן פולינומי. האלגוריתם של שור פותר את הבעיה השנייה במחשב קוונטי. עם זאת, הוא אינו מוכיח שאלגוריתם קלאסי יעיל אינו קיים. מחלקת בעיות ההכרעה הניתנות לפתרון במחשב קוונטי בזמן פולינומי, עם הסתברות לטעות שלכל היותר  $1/3$  בכל קלט, מסומנת BQP. מתקיימות ההכללות

$$P \subseteq BPP \subseteq BQP,$$

אך לא ידוע אם ההכללה האחרונה ממש. כלומר, השאלה

$$BPP \stackrel{?}{=} BQP$$

עדיין פתוחה, וכך גם השאלה אם  $P = BQP$ . ההשוואה למחלקה BPP חשובה במיוחד: אנחנו רוצים לדעת מה מוסיפות התופעות הקוונטיות גם כאשר למחשב הקלאסי מותר להטיל מטבע.

ומה לגבי תזת צ'רץ'-טיורינג המקורית? מודל החישוב הקוונטי הרגיל אינו סותר אותה: אפשר לסמלץ אותו במחשב קלאסי, אלא שהסימולציה הישירה עלולה לדרוש זמן מעריכי. השאלה המרתקת היא אם אפשר להימנע מן המחיר הזה.

תחום החישוב הקוונטי משלב תאוריה, ניסוי והנדסה, ומושך עניין גם הרבה מעבר לאקדמיה. התקדמות טכנולוגית, מימון ותחרות בין מדינות תורמים לעניין הזה, וכנראה גם ההייפ הטבעי עושה את שלו. כך או כך, זהו תחום שהוא תערובת יפהפייה של חישוב, מתמטיקה ופיזיקה, וכל תירוץ ללמוד עליו יתקבל בברכה.

## 2 מה נלמד בקורס?

הקורס מחולק לשבעה חלקים עיקריים. זהו המתווה המתוכנן; התכנים והקצב עשויים להשתנות מעט במהלך הסמסטר.

### פרק 1 – היכרות עם תורת הקוונטים

בפרק זה נציג את עקרונות היסוד של תורת הקוונטים הרלוונטיים למודל החישובי. נכיר את הקיוביט, המקבילה הקוונטית של הביט הקלאסי, את הפעולות שאפשר לבצע על קיוביטים ואת כללי המדידה שלהם. לעבודה בחישוב קוונטי יש אופי מתמטי, וכלי העבודה המרכזי הוא אלגברה לינארית; לכן נקדיש חלק מן הזמן גם ליישור קו מתמטי. לאורך הקורס נעבוד בעיקר עם המודל המתמטי המופשט. לצד זאת נשתמש בכמה ניסויים מחשבתיים ובתיאורים פיזיקליים ממושטים, כדי לפתח אינטואיציה לכללים החדשים.

הפרק מחולק ליחידות הבאות:

- יחידה 1 – מבוא (היחידה הנוכחית)

- יחידה 2 – היכרות ראשונה עם קיוביטים

- יחידה 3 – ניסוי הפצצה של אליצור-ויידמן

- יחידה 4 – המכפלה הטנזורית של מרחבים וקטוריים

- יחידה 5 – מערכות מרובות קיוביטים

## פרק 2 – תופעות ופרוטוקולים קוונטיים

בחלק זה נציג תופעות ופרוטוקולים קוונטיים מרתקים, עוד לפני שניגש לאלגוריתמים הפועלים על קלטים גדולים. ברוב היחידות נתמקד במערכות קטנות, לעתים של שניים או שלושה קיוביטים בלבד. כך נוכל להתרגל לכללים החדשים, ולעתים המוזרים, ולהתאמן בכלים המתמטיים הנדרשים, לפני שנעבור לחישובים על מספר רב של קיוביטים.

- יחידה 6 – משפט אי השכפול (No-cloning theorem)

- יחידה 7 – טלפורטציה קוונטית (Quantum teleportation)

- יחידה 8 – משחק CHSH, הפרת אי-שוויון בל וחסם צירלסון

- יחידה 9 – כסף קוונטי

## פרק 3 – מודל החישוב הקוונטי

לאחר שהכרנו כמה תופעות קוונטיות, נרצה להפוך את כללי המשחק למודל חישובי מסודר. נציג מעגלים קוונטיים ונראה כיצד אפשר לשלב בהם חישוב קלאסי באמצעות חישוב הפיך. נשאל אילו שערים דרושים כדי לבצע חישוב קוונטי כללי, וכיצד אפשר להסתפק באוסף סופי של שערים ולקרב בעזרתו פעולות אחרות. משפט סולוביי-קיטאיב יאפשר לנו להבין גם את המחיר החישובי של הקירוב.

נמקם את מודל החישוב הקוונטי ביחס למחלקות הסיבוכיות המוכרות, ונראה שכל בעיה במחלקה BQP ניתנת לפתרון במחשב קלאסי בזיכרון פולינומי, כלומר

$$\text{BQP} \subseteq \text{PSPACE}.$$

נכיר גם את עיקרון דחיית המדידה: אפשר להעביר את המדידות לסוף החישוב, גם כאשר תוצאותיהן משמשות לבחירת הפעולות הבאות, באמצעות הוספת קיוביטים והחלפת הבקרה הקלאסית בבקרה קוונטית. כך נקבל מודל נוח לעבודה, שבו החישוב מורכב מפעולות אוניטריות ובסופו מדידה.

- יחידה 10 – מעגלים קוונטיים וחישוב הפיך

• יחידה 11 – שערים אוניברסליים וקירוב: משפט סולוביי-קיטאיב

• יחידה 12 – BQP מוכלת ב-PSPACE

• יחידה 13 – עיקרון דחיית המדידה

#### **פרק 4 – אלגוריתמים קוונטיים בסיסיים**

כעת נתחיל לתכנן אלגוריתמים. נתחיל בבעיות פשוטות במודל האורקל, שבהן כבר אפשר לראות יתרון של חישוב קוונטי. האלגוריתמים של דויטש ושל ברנשטיין-וזיראני ימחישו כיצד שאילתה קוונטית אחת יכולה לחשוף מידע שלא ניתן להפיק משאילתה קלאסית אחת. משם נעבור לאלגוריתם של סיימון, שבו הפער במספר השאילות גדול הרבה יותר: אלגוריתם קוונטי משתמש במספר פולינומי של שאילות, בעוד שאלגוריתם קלאסי, גם אם מותר לו להטיל מטבע, זקוק למספר מעריכי במקרה הגרוע.

בסיום הפרק נחזור לאלגוריתמים של ברנשטיין-וזיראני ושל סיימון וננסה לזהות את הרעיון המשותף להם. המטרה אינה רק לצבור דוגמאות, אלא להבין כיצד תכונות של הפונקציה הנתונה באורקל מתורגמות למידע שאפשר לחלץ באמצעות התאבכות ומדידה. נקודת המבט הזו תלווה אותנו גם בדרך לאלגוריתם של שור.

• יחידה 14 – שאילתה אחת מספיקה: דויטש וברנשטיין-וזיראני

• יחידה 15 – האלגוריתם של סיימון

• יחידה 16 – האלגוריתמים של ברנשטיין-וזיראני ושל סיימון: סיכום ומבט מאחד

#### **פרק 5 – האלגוריתם של שור ובעיית תת-החבורה המוסתרת**

בפרק זה נגיע לאחת מגולות הכותרת של הקורס: האלגוריתם של שור לפירוק מספרים לגורמים. כדי להבין אותו נפתח את השפה המתמטית המתאימה: נכיר קרקטרים והתמרת פורייה על חבורות אבליות סופיות, ונבחן את הרעיון של סיימון בהקשר של חבורה ציקלית. כך נעבור מחיפוש מבנה חבוי במרחב של מחרוזות ביטים למציאת מחזוריות, שהיא מרכיב מרכזי באלגוריתם של שור.

בדרך נלמד גם על שברים משולבים, כלי קלאסי המאפשר לשחזר מידע מדויק מתוך קירוב מספרי. השילוב בין הדגימה הקוונטית לבין העיבוד הקלאסי של תוצאותיה יאפשר לנו להשלים את האלגוריתם. לבסוף נציג את בעיית תת-החבורה המוסתרת, Hidden Subgroup Problem (HSP), המספקת מסגרת משותפת לכמה מן האלגוריתמים שנלמד.

• יחידה 17 – קרקטרים והתמרת פורייה על חבורה אבלית סופית

• יחידה 18 – האלגוריתם של סיימון מעל חבורה ציקלית

• יחידה 19 – שברים משולבים

• יחידה 20 – האלגוריתם של שור

• יחידה 21 – בעיית תת-החבורה המוסתרת (HSP)

## פרק 6 – האלגוריתם של גרובר וחסמים תחתונים בסיבוכיות שאילתות קוונטית

עד כה ניצלנו מבנה מיוחד של הבעיה כדי להאיץ את החישוב. מה אפשר לעשות כאשר אין מבנה כזה, וכל מה שניתן לנו הוא לבדוק אם מועמד מסוים הוא פתרון? האלגוריתם של גרובר (Grover) מאפשר לחפש פתרון יחיד בין  $N$  מועמדים באמצעות  $O(\sqrt{N})$  שאילתות בלבד, במקום מספר שאילתות מסדר גודל  $N$  הנדרש בחיפוש קלאסי. זהו שיפור ריבועי, והוא מתקבל בלי להניח מבנה נוסף של מרחב החיפוש.

אבל האם אפשר לעשות טוב יותר? כדי לענות על שאלות כאלה דרושים חסמים תחתונים, התקפים לכל אלגוריתם קוונטי אפשרי. נלמד שתי שיטות מרכזיות להוכחת חסמים כאלה: שיטת היריב והשיטה הפולינומית. כך נכיר גם את המגבלות של החישוב הקוונטי, ונבין מדוע השיפור של גרובר בחיפוש בלתי מובנה הוא מיטבי.

• יחידה 22 – האלגוריתם של גרובר

• יחידה 23 – חסמים תחתונים: שיטת היריב

• יחידה 24 – חסמים תחתונים: השיטה הפולינומית

## פרק 7 – אלגוריתמים ספקטריים קוונטיים

בפרק האחרון נעבור לאלגוריתמים המבוססים על הערכים העצמיים והווקטורים העצמיים של אופרטורים. נתחיל באלגוריתם הערכת פאזה, המאפשר לאמוד את הפאזה של ערך עצמי של אופרטור אוניטרי. זהו כלי מרכזי שאפשר לשלב באלגוריתמים אחרים, והוא ייתן לנו גם נקודת מבט נוספת על הרעיונות שפגשנו קודם.

לאחר מכן נעסוק בסימולציה המילטונית: מימוש ההתפתחות בזמן של מערכת קוונטית המתוארת על ידי המילטוניאן. במונחים מתמטיים, עבור מטריצה הרמיטית  $H$  נרצה לממש בקירוב את האופרטור  $e^{-iHt}$ . לבסוף נלמד את אלגוריתם HHL למערכות משוואות לינאריות. כאן יהיה חשוב להבין גם מה האלגוריתם מקבל ומה הוא מחזיר: בתנאים מתאימים הוא מכין מצב קוונטי המקודד את וקטור הפתרון, ואינו מדפיס את כל רכיבי הפתרון כרשימה קלאסית. ההבחנה הזו חיונית להבנת היתרון האלגוריתמי ולזיהוי הבעיות שבהן אפשר לנצל אותו.

• יחידה 25 – אלגוריתם הערכת פאזה

• יחידה 26 – סימולציה המילטונית

• יחידה 27 – אלגוריתם HHL

למי שירצה להמשיך, קורס בתורת האינפורמציה הקוונטית הוא המשך טבעי לקורס הזה: הוא מעמיק בשאלות של ייצוג מידע, העברתו ועיבודו בעולם הקוונטי.