**דרישות מערכת לפרויקט השוואת מספרים מבוסס תהליכים**

**דרישות חומרה**

* **מעבד:** מעבד מודרני מרובה ליבות (למשל, Intel Core i5/i7, AMD Ryzen 5/7)
* **זיכרון:** לפחות 2 ג'יגה-בייט RAM (4 ג'יגה-בייט ומעלה מומלץ לביצועים טובים יותר)
* **אחסון:** שטח דיסק מינימלי נדרש (פחות מ-100 MB)
* **מערכת הפעלה:** תואמת עם מערכות הפעלה עיקריות (Windows, Linux, macOS)

**דרישות תוכנה**

* **קומפיילר:**
  + GCC (GNU Compiler Collection) גרסה 9.0 ומעלה
  + Clang גרסה 10.0 ומעלה
  + Microsoft Visual Studio 2019 ומעלה (למשתמשי Windows)
* **ספריות:**
  + ספריית C++ סטנדרטית (כוללת <iostream>, <thread>, <mutex>, <condition\_variable>, <queue>, <chrono>, <random>)

**בנייה והרצה**

1. **בניית הפרויקט:**
   * ל-GCC/Clang:

g++ -std=c++17 -o number\_comparison number\_comparison.cpp

* + ל-Visual Studio:
    - פתח את הפרויקט ב-Visual Studio ובנה באמצעות ה-IDE.

1. **הרצת התוכנית:**
   * הרץ את הבינרי שנבנה:

./number\_comparison

**תלותים**

* הפרויקט תלוי בספריות C++ סטנדרטיות. ודא שהקומפיילר שלך תומך ב-C++17 או מאוחר יותר.

**פרמטרי קונפיגורציה**

* **message\_delay\_seconds:** זמן המתנה בשניות בין שליחת הודעות חדשות (ברירת מחדל: 1 שניה)
* **check\_interval\_seconds:** זמן המתנה בשניות שבו תהליך 3 בודק את התורים (ברירת מחדל: 5 שניות)
* **max\_initial\_random\_number:** הגבול העליון לערך ההתחלתי של מספרים רנדומליים (ברירת מחדל: 10)
* **max\_increment:** הגבול העליון להגדלת המספרים ליצירת המספר הבא (ברירת מחדל: 10)
* **increment\_value:** ערך ההגדלה לכל מספר (ברירת מחדל: 10)

**התנהגות צפויה**

* **תהליך 1 ותהליך 2:**
  + מקבלים ושולחים מספרים רנדומליים בזמנים קבועים (בהגדרה של message\_delay\_seconds), ומוסיפים מספרים לתורים שלהם.
  + התהליכים נמצאים במצב המתנה (wait) כאשר הם לא צריכים לשלוח הודעות, וממתינים להתראה לפני שליחה של הודעה חדשה.
* **תהליך 3:**
  + בודק תקופתית את התורים של תהליך 1 ותהליך 2 (בהגדרה של check\_interval\_seconds).
  + משווה בין המספרים בתורים ומדפיס מידע על מספרים משותפים או מוסרים.
  + התהליך ממתין להתראות על הודעות חדשות בתורים, ומשווה בין המספרים רק כאשר יש הודעות חדשות לבדיקה.

**פתרון בעיות**

* **בעיות קימפול:** ודא שאתה משתמש בקומפיילר תואם ושה-C++17 או מאוחר יותר מופעל.
* **בעיות הרצה:** ודא שאין בעיות במשאבי המערכת ושהתוכנית יש לה את ההרשאות המתאימות להרצה.

**תיאור עקרוני של התוכנית:**

* **ניהול תורים:**
  + כל תהליך משתמש בתור משלו כדי לשמור על המספרים שהוא שולח. תורים אלה מנוהלים עם מנעולים (mutex) כדי למנוע גישה מקבילה לא מבוקרת לנתונים.
* **השיהיה והמתנה:**
  + התהליכים מבצעים השהיות (sleep) בפרקי זמן מוגדרים. תהליך 3 משתמש במשתנה תנאי (condition\_variable) כדי לחכות להתראות על הודעות חדשות בתורים.
* **שימוש בתהליכים מקביליים:**
  + התוכנית עושה שימוש בתהליכים מקביליים כדי לשפר את ביצועי התוכנית ולהשיג עבודה מקבילית.
* **השוואה והדפסה:**
  + תהליך 3 עורך השוואות בין המספרים בתורים ומדפיס מידע על המספרים המשותפים והמספרים המוסרים.
* **התוכנית מדגימה כיצד ניתן להשתמש בתהליכים מקביליים ובתיאום בין תהליכים כדי לבצע פעולות סימולטניות, כגון שליחה והשוואה של מספרים רנדומליים בזמן אמת.**