

קצת על ההיסטוריה של C

שפת C הומצאה ע"י Dennis Ritchie במעבדות Bell בשנים 1969-1973. השפה פותחה מתוך שפה קיימת בשם B (שהומצאה ע"י Ken Thompson) שבעצמה פותחה מתוך שפה הנקראת BCPL (שהומצאה ע"י Martin Richards). הפיתוח של C נעשה בשילוב הדוק עם פיתוח מערכת ההפעלה UNIX.

בסוף שנות השישים עבדו צוותי פיתוח במעבדות Bell על מערכת הפעלה חדשה בעלת יומרות גדולות בשם Multics. הפרויקט היה מיזם משותף של מעבדות Bell ומספר גורמים נוספים. בשלב מסוים הגיעו להבנה שפרויקט המערכת החדשה התקדם לאט מדי ונהיה יקר מדי. קבוצת פיתוח לא פורמלית שהובלה בעיקר ע"י Ken Thompson וכללה גם את Dennis Ritchie החלה לחפש אלטרנטיבות. קבוצת הפיתוח נאלצה להתמודד עם אילוצי חומרה מגבילים אפילו בייחס לאותה תקופה: הם עבדו על מחשב בשם PDP-7 של חברת DEC (שמאוחר יותר נקראה digital). הזיכרון של המחשב זה הוא רק 18K של מילים בגודל 18 ביט – פחות מכמות הזיכרון הדרושה למסמך זה.

בשלב הראשון נכתבה הגרסה הראשונה של מערכת ההפעלה UNIX בשפת האסמבלי של ה-PDP-7. השם UNIX היה במקור UNICS - מעיין קריצה שהמציא Brian Kernighan לשם המערכת הקודמת Multics. צוות הפיתוח היה מעוניין מאד שמערכת ההפעלה החדשה תהיה כתובה בשפה עילית ולא באסמבלי כפי שהייתה כתובה. הסיבה לכך הייתה בייחוד הרצון בבהירות הקוד וקלות התחזוקה וההרחבה שלו. בשלב מאוחר יותר נוספה סיבה חשובה נוספת: האפשרות להעביר את הקוד לפלטפורמות חדשות – פורטביליות. קבוצת הפיתוח החלה לחפש שפה בה תיכתב המערכת החדשה. השפה PL/I שבה הכתבה Multics לא הייתה לטעמם של המפתחים. אחרי ניסיון קצר ולא מוצלח עם שפת התכנות Fortran, החליט Thompson ליצור שפת תכנות משלו. השפה החדשה הייתה פיתוח של שפת תכנות קיימת שנקראה BCPL והותאמה במיוחד לדרישות הזיכרון המגבילות שעמן נאלץ להתמודד. Thompson קרא לשפתו החדשה B כנראה כקיצור של BCPL או על שם שפה אחרת שיצר ונקראה Bon.

BCPL היא שפה קומפקטית ייחסית ו B היא מעיין כיווץ שלה בכדי שתתאים לאילוצי הזיכרון הקשים. חשוב לציין שלכתוב קומפיילר לשפה B הייתה משימה קלה בהרבה מאשר לכתוב קומפיילר לאחת מהשפות העיקריות באותה תקופה: PL/I, Fortran או Algol 68. למרות ששפות אלו השפיעו על הפיתוח של B אחר כך של C, הגישה הכללית של צוות הפיתוח הייתה הרבה יותר מינימליסטית. לאחר זמן מה קיבל צוות הפיתוח מחשב מתקדם יותר – PDP-11 ונכתבה גירסת UNIX עבורו, בשפת האסמבלי שלו. האפליקציות שנכתבו למערכת החדשה נכתבו כבר ב-B. השפה B עדיין לא הייתה בשלה לכך שמערכת ההפעלה תכתב מחדש בה. אחד החסרונות החשובים היה חסרונם של טיפוסי משתנים בשפה.

בשלב זה החל Dennis Ritchie לכתוב קומפיילר חדש ל B שייצר קוד יעיל יותר. תוך כדי עבודה, שינה Ritchie קצת את השפה והוסיף טיפוסי משתנים. לשפה אחרי השינוי הקל קרא Ritchie בשם NB עבור New B. השפה NB התקיימה זמן קצר ביותר מכיוון שמהר מאד שכלל Ritchie את מנגנון הטיפוסים והגיע למסקנה שלאחר שינויים אלה כבר מגיע לשפה שם משל עצמה. הוא קרא לשפה החדשה C לציין את ההתקדמות מ-B, או בסדר האלפביתי או באותיות שם השפה המקורית: BCPL.

השפה עברה התפתחות אינטנסיבית עד שבשנת 1973 הייתה בשלה לכך שהגרעין של UNIX יכתב מחדש ב-C. בשנים הבאות הועברה C לפלטפורמות אחרות והחלה לצבור פופולריות. בשנת 1978 הוציאו Dennis Ritchie ו-Brian Kernighan את ספרם "The C Programming language" שהפך למעין תקן לא-רשמי של השפה. התקן הרשמי פורסם יותר מעשר שנים מאוחר יותר. הספר ידוע גם כ-K&R או כ"ספר הלבן" (בלי קונטציות מנדטוריות) ונחשב בעיני רבים כספר מצוין. במהדורתו הראשונה תיאר הספר גרסה מוקדמת של C (גרסה זו נקראת לעתים K&R C) ובמהדורתו הנוכחית הוא מתאר את השפה לפי התקן של ANSI (המכון הלאומי האמריקאי לתקנים). מאז שנת 1978 השפה התפתחה אבל לא עברה שינויים מרחיקי לכת. בשנת 1999 נקבע ל C תקן חדש המכונה C99 המוסיף לשפה מאפיינים רבים. לא כל הקהילה התכנותית ממהרת להפנים ולהתיישר לפי התקן החדש.

מדוע C הצליחה כל כך

שפת C היא אחת משפות התכנות המצליחות ביותר בהיסטוריה, אולי אפילו המצליחה מכולן. יוצרי השפה לא ציפו בשום אופן להצלחה כזו והיא באה להם בהפתעה גמורה. מפתיע ששפה שפותחה ללא יומרות, רזה בהרבה מהשפות המרכזיות באותה תקופה, הפכה לדומיננטית כל כך וגרמה לחברותיה להיות לא רלוונטיות. ניתן לנסות למנות מספר סיבות להצלחה זו:

- **הפשטות של C** הקלה על כתיבת קומפילרים לשפה ואפשרה להשתמש ב C על הרבה פלטפורמות. גרעין השפה הוא קטן וחלק גדול מהמשימות מבוצעות ע"י ספריה חיצונית הכתובה בשפה עצמה. גם עובדה זו מקלה על כתיבת הקומפילר. אותה פשטות הקלה גם בכתיבת קומפילרים המייצרים קוד יעיל וחסכוני. בהרבה מיקרים קוד שנכתב ב-C יעיל כמעט כמו קוד שנכתב בשפת מכונה. הפשטות גם מקלה על תהליך הלימוד של השפה ומושכת מתכנתים ללמוד אותה.

- **C מהווה פשרה מוצלחת בין שפה עילית לשפת סף**. בעד ש-C מאפשרת שליטה במכונה הדומה לשליטה הנתונה בידי מתכנת בשפת מכונה, היא עילית מספיק בשביל שאפשר יהיה להעביר קוד בין פלטפורמות. לפעמים מכנים את C כשפת ביניים במקום שפה עילית. כינוי אחר הוא "אסמבלי פורטבילית".

- **ההצלחה של מערכת ההפעלה UNIX** תרמה ללא ספק להצלחה של C. ב-UNIX, אולי מערכת ההפעלה המשפיעה ביותר בהיסטוריה, כל הממשק לתוכנות (אוסף ה-systemcalls) מנוסח ב C וכל המערכת והאפליקציות שמרכיבות אותה כתובות ב-C. קשה להיות מתכנת בסביבת UNIX בלי לדעת C.

- **שפה כלשהי הייתה חייבת להצליח**. העולם אוהב סטנדרט יחיד בכדי לשפר שיתופי פעולה בין אנשים. ריבוי שפות תכנות מקשה על העברת קוד בין אנשים שונים ותחזוקת קוד לאורך שנים. יש מעין כוח המושך לכיוון בחירת סטנדרט אחיד שהוא נוח מעצם היותו יחיד ולא דווקא בגלל היותו הטוב ביותר (ראה ערך microsoft).

נראה שסביבת הפיתוח הספרטנית בה צמחה C ייצרה שפה פשוטה, נקייה ותמציתית שהתאימה לצרכי התכנות של העולם. אולי השפה הצליחה יותר בזכות מה שאין בה מאשר מה שיש בה. למרות ההצלחה הגדולה, C רחוקה מלהיות מושלמת. חלק מהפגמים תוקנו ב-C++, שפה שפותחה מתוך C.

Dennis Ritchie המודע לבעיות השפה ומופתע מהצלחתה, מתאר זאת כך:

"C is quirky, flawed, and an enormous success."