

הספריה התקנית - אלגוריתמים

בספריה התקנית של C++ יש 105 אלגוריתמים, נכון לשנת 2017. אנחנו צריכים להכיר את כולם.

למה? - כי האלגוריתמים פותרים בעיות מאד נפוצות בתיכנות. אם לא נכיר את כולם, אנחנו עלולים לנסות לממש אותם בעצמנו תוך כדי פרויקט אחר. ואז, אנחנו כנראה נממש אותם בחיפזון, בלי בדיקות, בצורה לא יעילה ועם באגים. בנוסף, האלגוריתמים נכנסו לתקן של השפה, ולכן מתכנתים בכל העולם משתמשים בהם ומבינים אותם, ומצפים שגם מתכנתים חדשים יכירו אותם וישתמשו באותה שפה.

איך אפשר לזכור 105 אלגוריתמים? - זה לא קל, אבל אנחנו נחלק אותם לקבוצות הגיוניות (נע"פ המצגת המעולה של יונתן בוקארה - ראו קישור למטה).

שאילות - queries

ערימות - heaps

ערימה היא עץ בינארי מלא, שבו הערך בכל צומת גדול יותר מהערךים בכל צאצאיו (בניגוד לעץ-סידור בינארי, שבו הערך בכל צומת גדול מהערךים של הצאצאים משמאלו וקטן מהערךים של הצאצאים מימינו). כיוון שהעץ מלא (כמעט לכל צומת פנימי יש שני ילדים), אפשר לייצג אותו כמערך. האלגוריתמים הפועלים על ערימה, מקבלים כקלט שני איטרטורים - לתחילת המערך ולסופו.

הפונקציה generate משמשת ליצירה אוטומטית של ערכים; הפונקציה bind משמשת לקשירת ארגומנט לפונקציה. ראו דוגמה בתיקה 7.

האלגוריתמים בספריה התקנית מקבלים כקלט זוג איטרטורים ולא מיכל (זאת בניגוד לג'אבה. בג'אבה יש אלגוריתם סידור נפרד עבור List, עבור מערך של תוים, מערך של מספרים וכו'...). דוגמאות לאלגוריתמים (ראו בתיקוד הספריה; ראו דוגמאות בתיקה 6):

- סידור - sort
- מיזוג מערכים מסודרים - merge
- העתקה - copy

סוגי איטרטורים

כל אלגוריתם דורש איטרטור ברמה מסויימת (ראו טקסט קודם על איטרטורים).

לדוגמה, האלגוריתם sort עובד על איטרטורים מסוג RandomAccess. לכן אפשר להשתמש בו לסידור וקטור וגם מערך פרימיטיבי.

אבל, אי אפשר להשתמש באלגוריתם sort לסידור list, כי האיטרטור שלה הוא מסוג Bidirectional (לא תומך למשל בפעולת חיסור).

(מה עושים? משתמשים בשיטת sort המיוחדת של list; ראו תיקיה 6).

הודעות שגיאה

אחד הקשיים העיקריים בעבודה עם STL הוא הודעות השגיאה. למשל, אם ננסה להריץ את אלגוריתם sort על list, לא נקבל הודעה פשוטה שאומרת "אי אפשר להריץ sort על list", אלא הודעה ארוכה ומסובכת הנכנסת לפרטי התבניות בספריה התקנית (ראו דוגמה בתיקיה 6). כדי לפענח את הודעת השגיאה, צריך לחפש את ה-note המפנה לשורה בקוד שלנו, ומשם לנסות להבין מה הבעיה.

ישנן ספריות המנסות לתת הודעות שגיאה משמעותיות יותר, למשל STLfilt, boost - לא ניכנס לזה בקורס הנוכחי.

ספריות נוספות

בנוסף לספריה התקנית, יש ספריות נוספות. המקובלת ביותר היא boost היא "כמעט" תקנית - הרבה מהדברים בספריה התקנית התחילו את דרכם ב-boost, השתכללו והשתפרו, עד שבסוף נכנסו לספריה התקנית. לכן, אם חסר לכם משהו בספריה התקנית - נסו לחפש ב-boost.

מקורות

- Jonathan Boccara, "105 algorithms in less than an hour", CPPCON 2018, <https://youtu.be/2olsGf6JIku>
- Sean Parent, "C++ Seasoning", GoingNative 2013, <https://www.youtube.com/watch?v=qH6sS0r-yk8>
- Jonathan Boccara, "Is for_each obsolete? (no)" https://www.fluentcpp.com/2018/03/30/is-stdfor_each-obsolete
- מצגות של אופיר פלא.
- Peter Gottschling, "Discovering Modern C++", chapter 4
- Marius Bancila, "Modern C++ Programming Cookbook", chapter 5
- תיעוד הספריה התקנית: <http://www.cplusplus.com/reference/stl>

ברוך ה' חונן הדעת

סיכום: אראל סגל-הלוי.