

מבוא מורחב למדעי-המחשב ~ תרגיל בית 2

שחר פרץ

27 ביוני 2024

1

סעיף ב'

להלן זמני הריצה עבור $c = 2, 3$:

c-value	2	3
time (s)	$8.4658 \cdot 10^{-5}$	0.00566

נתבקשנו לכתוב, מה קורה בעבור $c = 5$. למרבה הצער, המחשב שלי לא יכול להריץ את הקוד במימוש הנוכחי כדי למצוא זאת, ככל הנראה בגלל העובדה שלפי ויקיפדיה זהו מספר בן 8 ספרות, כלומר נצטרך לעבור עליו וכל המספרים שלפניו כדי להבין שהוא מספר משוכלל.

סעיף ד'

נייצג הפלט של abundant_density עבור מספר ערכים:

n	50	500	2500	5000	7500	10000
output	0.18	0.242	0.2468	0.2478	0.2490666	0.2488

כלומר, נראה שכן בשאיפה עבור ערכים גדולים יותר ויותר הערכים קרבים לחסמים, אם כי באופן מוזר תחילה הם מתחתיהם ואז הם מעליהם.

המשך בעמוד הבא

סעיף ג'

נקבל את התוצאות הבאות:

```
1 >>> monty_hall(True, 10000)
2 0.6647
3 >>> monty_hall(False, 10000)
4 0.3217
```

לכן, האסטרטגיה של לבחור להחליף את הדלת עדיפה – היא תביא ניצחון בבערך $\frac{2}{3}$ מהמקרים, בעוד לא להחליף דלת יביא לניצחון ב- $\frac{1}{3}$ מהמקרים.

סעיף ה'

כדי לוודא את נכונות הפונקציה שמומשה בסעיף ד', נריץ 10^4 בעבור הקלט `st = "abc"`. נציב את התוצאות בטבלה:

word	1663	1675	1667	1705	1636	1654
count	abc	bac	cab	acb	cba	bca

נטען, שהפונקציה תקינה כי נצפה בממוצע לקבל $666\frac{2}{3} = \frac{1000}{6}$ בעבור כל אחד מהם (תחת ההנחה שהפרמוטציות מתחלקות באופן שווה), ואכן אלו המספרים שנקבל (תחת שגיאת המדידה).

המשך בעמוד הבא

סעיף ה'

יהי N מספר בבסיס b בעל d_b ספרות, ויהי בסיס c . נרצה להוכיח כי ייצוג N באמצעות c ספרות ידרוש לכל היותר $\lceil d_b \cdot \log_c b \rceil$ תווים. הוכחה. טענת עזר: ידוע שעבור מספר כלשהו x בבסיס b' עם d' ספרות יתקיים $x < b'^{d'}$ ולכן $\lceil \log_{b'} x \rceil \leq d'$. נשתמש בחסם זה. נחזור להוכחה. יתקיים $N < b^{d_b}$ לפי חסמים מהרצאה. אזי, כמות הספרות לייצוג N בבסיס c , נסמנה d_c , לפי החסם לעיל, תקיים:

$$d_c \leq \lceil \log_c N \rceil \leq \lceil \log_c b^{d_b} \rceil = \lceil d_b \cdot \log_c b \rceil$$

■

מטרנזיטיביות יחס הסדר על הטבעיים, נקבל את הדרוש.

המשך בעמוד הבא

הסבר:

1. ב-breakpoint 1, נייצר את המשתנים ונשלח פוינטרים למיקום המתאים בזכרון. לאחר שנפנה את השם y לערך החדש שלו, 20, הערך הישן שלו "cs" ישאר המרחב הזכרון ולא ימחק, אם כי לא יפנה אליו שום אובייקט.
2. ב-breakpoint 2, עליה נדבר לפני 2 מבוצע לאחר 3, תתווסף פונקציה לזכרון (שבפיאצה נאמר שלא צריך לציין אותה במקרה הזה), והאופרציה + תיצור רשימה חדשה בזכרון - שרשור של $lst1$ ו- $lst2$, ואל התוצאה יפנה השם $lst1$ תוך מחיקת ההפניה הקודמת.
3. ב-breakpoint 2, הפונקציה $what()$ תפתח scope חדש, כלומר מרחב שמות חדש ייוצר. ראוי לציין שלמשתנה הלוקאלי x ישוייך ערך חדש בהתאם לחישוב 8-10, $lst1$ (הפניה לוקאלית לאותה הפניה של $lst1$ הגלובאלי) שיפנה לאותה הרשימה שקיבל כקלט, ישנה את הערך הראשון בו ל-"i" - שם חדש בזכרון. לבסוף מתבצעת השמה לשם $lst2$ לרשימה חדשה בזכרון, שגם מוחזרת.
4. לבסוף, ב-breakpoint 4 מרחב השמות הלוקאלי של $what$ לא רלוונטי יותר כי יצאנו מה-scope של הפונקציה, וערך ההחזרה שלה ישוייך לשם $lst3$, כלומר הוא יפנה אליו.

ציור:

