

מטלת מנחה (ממ"ן) 15

הקורס: 20441 - מבוא למדעי המחשב ושפת Java

חומר הלימוד למטלה: יחידה 11 נושא המטלה: רשימות מקושרות

מספר השאלות: 1 משקל המטלה: 5 נקודות

סמסטר: 2021 מועד אחרון להגשה: 23.1.2021

בתרגיל זה תתמודדו עם בעיה פרקטית – אנו יודעים שהמספרים אותם ניתן לשמור בתאי זיכרון מסוג `int` או `long` מוגבלים בגודלם (בגלל גודל הזיכרון). אולם לפעמים נצטרך להשתמש באפליקציות מסוימות במספרים גדולים מאוד, כדוגמת המספר הבא:

233,128,765,555,234

מספר זה בן 15 ספרות לא יכול להישמר בתא זיכרון בודד מסוג `long` בגלל גודלו הרב. דרך אחת להתמודד עם שמירת מספרים גדולים מאוד היא באמצעות רשימה מקושרת. נוכל להגדיר רשימה מקושרת שכל איבר בה יכיל ספרה אחת מהמספר, כך שהמספר עצמו לא ישמר כמספר אלא כאוסף ספרותיו אחת אחרי השנייה. בתרגיל זה תגדירו רשימה מקושרת לשמירת מספרים גדולים מאוד, ותגדירו כמה פעולות שימושיות על הרשימה.

אנו מתייחסים במטלה זו למספרים אי-שליליים בלבד (כלומר חיוביים ואפס).

שימו לב שהרשימה המקושרת חייבת להיות חד-סטרית.

שאלה 1 (100%)

הגדירו את המחלקה `BigInteger`. מחלקה זו תייצג רשימה מקושרת של ספרות, שייצגו מספר גדול מאוד. תוכלו להשתמש במחלקה `IntNode` שהוצגה בהרצאות כדי להגדיר את החוליה הבסיסית של הרשימה. כמו כן, תוכלו להתבסס על ההגדרה של המחלקה `IntList` שהוצגה אף היא בהרצאות, כדי לבנות את המחלקה `BigInteger`.

שימו לב – השקיעו מחשבה בייצוג המספר. כיוון שמדובר ברשימה מקושרת יש לכם שתי אפשרויות – לייצג את המספר מהסוף להתחלה (כלומר, האיבר הראשון ברשימה יכיל את ספרת האחדות במספר) או מההתחלה לסוף (כלומר, האיבר הראשון ברשימה יכיל את הספרה השמאלית ביותר במספר). חשבו היטב איזה ייצוג עדיף מבחינת המעברים על הרשימה (רמז – קחו בחשבון שהפעולות החשבוניות על מספרים שלמים מתחילות בד"כ בחישוב על הספרות הימניות יותר...).

אתם רשאים להחליט על המבנה הפנימי של המחלקה בעצמכם, אולם הממשק של המחלקה חייב להכיל רק (ואך ורק) את השיטות הבאות:

בנאים

כתבו למחלקה שלושה בנאים :

1. בנאי ריק שמאתחל את הרשימה למספר 0. כלומר, רשימה שמכילה איבר אחד שספרתו היא 0.
2. בנאי שמקבל כפרמטר מספר מסוג long ומאחסן אותו בפורמט של רשימה.
3. בנאי העתקה.

שיטות

מאחר והמחלקה מייצגת מספרים שלמים, נרצה לבצע בקלות פעולות חשבון שונות על המספרים. לכן עליכם להגדיר את השיטות הבאות :

- `public String toString()`

השיטה צריכה להחזיר כמחרוזת תווים את המספר שמיוצג ברשימה. המחרוזת תכיל את המספר בצורה הרגילה שלו, כלומר, הספרה המשמעותית ביותר תופיע ראשונה, ואחריה שאר הספרות.

חובה לממש את השיטה בזמן $O(n)$ כאשר n הוא מספר הספרות במספר. אסור ליצור רשימות חדשות, מערכים או להפוך את הרשימה למספר. ממשו את השיטה הזאת ברקורסיה.

- `public int compareTo (BigInteger other)`

שיטת השוואה בין שני מספרים גדולים. אם המספר עליו מופעלת השיטה קטן מהמספר שהתקבל כפרמטר, יוחזר הערך -1, אם המספר עליו מופעלת השיטה גדול מהמספר שהתקבל כפרמטר, יוחזר הערך 1, אם המספרים שווים יוחזר הערך 0.

- `public BigInteger addBigInteger (BigInteger other)`

שיטת החיבור של שני מספרים גדולים. החיבור מחזיר כתוצאה אובייקט חדש מסוג `BigInteger`.

- `public BigInteger addLong (long num)`

שיטת החיבור של אובייקט מסוג מספר גדול ועוד מספר `long`. החיבור מחזיר כתוצאה אובייקט חדש מסוג `BigInteger`.

- `public BigInteger subtractBigInteger (BigInteger other)`

שיטת החיסור של שני מספרים גדולים. החיסור הוא של המספר הגדול מהשניים פחות המספר הקטן. מחזיר כתוצאה אובייקט חדש מסוג `BigInteger`. אי אפשר להניח מי מהשניים גדול. יש לבדוק זאת (כדי שהתוצאה לא תהיה מספר שלילי).

- `public BigInteger multBigInteger (BigInteger other)`

שיטת הכפל של שני מספרים גדולים. הכפל מחזיר כתוצאה אובייקט חדש מסוג `BigInteger`.

הערה כללית לגבי השיטות חיבור, חיסור וכפל. השיטות לא ישנו את האובייקט עליו הן הופעלו ולא את האובייקט שהועבר כפרמטר, אלא יצרו `BigInteger` חדש שיכיל את פתרון התרגיל.

כל הפעולות חייבות להיות יעילות ככל שניתן מבחינת המעברים על הרשימה (כלומר – אם אפשר להסתדר רק עם מעבר אחד על הרשימה, עשו זאת. אין טעם להשקיע במעברים מיותרים על הרשימות).

הערה חשובה – בכל השיטות בתרגיל **אסור** להפוך את המספר שמיוצג על-ידי הרשימה למספר אמיתי, לבצע עליו את הפעולות ואז להפוך אותו חזרה לרשימה (מה הסיבה?). כמו כן **אסור** להשתמש במערכים בתרגיל זה.

אנחנו כתבנו את הממשק הפומבי ועליו לא ניתן להוסיף שיטות, אולם ניתן להוסיף שיטות פרטיות לנוחות וקריאות המחלקה.

שימו לב לכל מקרי השגיאה האפשריים!

כתבו (כחלק מהתיעוד) מה סיבוכיות הזמן וסיבוכיות המקום של כל שיטה שכתבתם. הקפידו על יעילות השיטות שכתבתם!

שימו לב:

1. אסור להשתמש במחלקות מוכנות כבר של Java.
2. מותר ורצוי להשתמש במחלקות שניתנו בהרצאה ונמצאות `IntNode`, `IntList` בחוברת השקפים.
3. עליכם לכתוב תיעוד API ותיעוד פנימי מלא.

הגשה

1. הגשת הממ"ן נעשית בצורה אלקטרונית בלבד, דרך מערכת שליחת המטלות.
2. הקפידו ששמות השיטות יהיו בדיוק לפי הוראות הממ"ן.
3. עליכם להגיש את הקובץ `BigInteger` עטפו אותו בקובץ `zip` ושלחו. אין לשלוח קבצים נוספים.

ב ה צ ל ח ה