



רשימות ומחלקות

עד עכשיו השתמשנו ב nodes שהערך שלהם היה מספר. ניתן להשתמש ב-
Node<T> כש T הוא מחלקה מטיפוס כלשהו

בכונן T נמצא הפרוייקט Student . בפרוייקט 4 מחלקות

Node – 1

Grade – 2 – שתכונותיה מקצוע וציון

Student - 3 – שתכונותיה שם, ת.ז. מספר טלפון, שנת לידה ורשימת ציונים

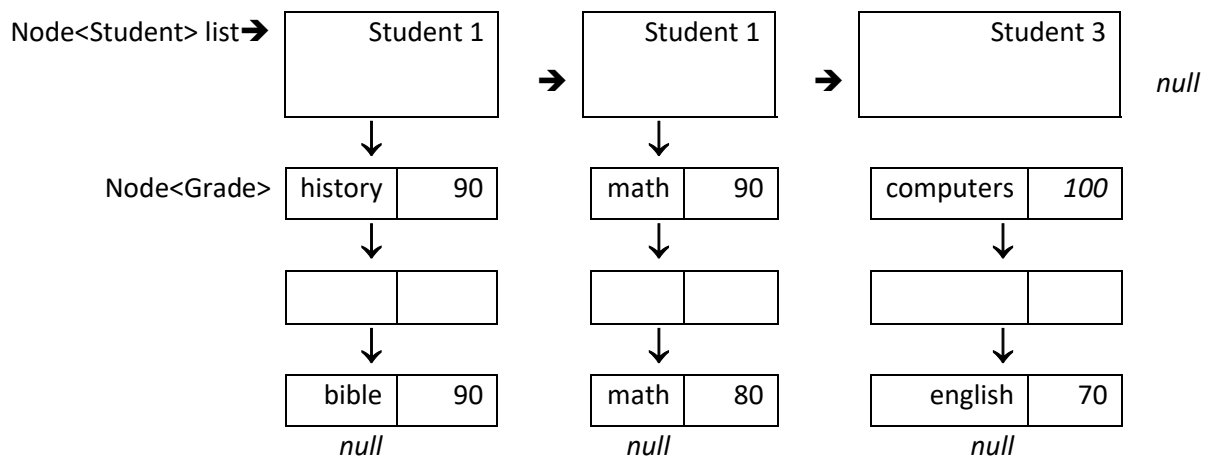
– runStudent – 4

- קולטת 3 סטודנטים
- לכל אחד מהם קולטת ציונים ל 3 מקצועות
- מדפיסה את פרטי הסטודנט – כולל הציונים חישוב ממוצע
- מחשב את ממוצע של כל ממוצעי הסטודנטים

נשים לב כי התכונה grades במחלקה Student היא בעצם הפנייה לרשימה של nodes שהערך של כל אחד מהם הוא מטיפוס Grade – Node<Grade>

גם בתוכנית הראשית יצרנו רשימה של nodes. הערך של כל node בתוכנית הראשית הוא מטיפוס Student. – Node<Student>

כך שלמעשה יש לנו רשימת סטודנטים שכל אחד מהם מצביע על רשימת ציונים.



ממש את הפעולות
הבאות במחלקה בשם
ListsAndClasses

1 תרגילים

1.1 משחק שש-בש

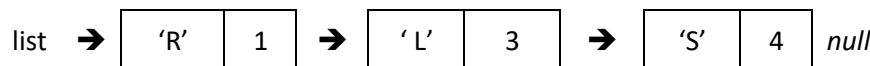
הטלת קובייה מוגדרת על ידי המחלקה Dice שמכילה 2 תכונות מסוג integer, roll1 ו roll2.
כתוב פעולה חיצונית בשם backgammon המקבלת רשימה שמכילה את נתוני הטלות הקובייה במשחק שש-בש.
כל חולייה מכילה הטלה של שתי קוביות במהלך אחד של המשחק (עצם מהמחלקה Dice). על הפעולה להחזיר את המספר שהוטל הכי הרבה פעמים. הנח שיש רק מספר אחד. אם הרשימה ריקה יוחזר 0.

1.2 רצף תווים

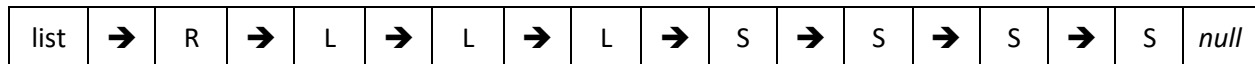
כתוב פעולה חיצונית בשם unpack המקבלת רשימה שכל חולייה בה מכילה עצם מסוג CharNum שלו 2 תכונות: תו (tav) ומספר שלם (num) והבנאי שלו CharNum(char tav, int num). הפעולה מחזירה רשימה המכילה רצפים של תווים זהים. עבור כל איבר ברשימה המקורית יכנסו לרשימה החדשה num תווים של tav ברצף. כל רצף תווים יתאים לאיבר מהרשימה המקורית

לדוגמא:

עבור הרשימה:



תחזיר הפעולה:



1.3 מתוך בגרות

המחלקה מסלול אוטובוס BusRoute מתארת מסלול של קו אוטובוס. כל מסלול מכיל מספר מסויים של תחנות, לפחות שתיים. כל תחנה מיוצגת על ידי זוג מספרים המייצג את מקומה במישור. כל תחנה מופיעה במסלול פעם אחת. לדוגמא מסלול בן 5 תחנות המתחיל בתחנה (0, 2) ונגמר בתחנה (5, 0)

$(0, 2) \rightarrow (1, 4) \rightarrow (5, 4) \rightarrow (3, 1) \rightarrow (5, 0)$

הנח שנתונה המחלקה station שתכונותיה שני מספרים שלמים x ו y המציינים את מיקומה של התחנה במישור.

במחלקה Station הוגדרו 2 הפעולות:

- פעולה בונה Station(int x, int y)
 - פעולה שכותרת double distance(station other)
- הפעולה מקבלת תחנה other ומחזירה את המרחק בין התחנה הנוכחית לתחנה other – לפי הנוסחה:

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

לפניך ממשק המחלקה מסלול אוטובוס BusRoute

BusRoute(station first, station second)	פעולה בונה המקבלת 2 תחנות ויתרת מסלול אוטובוס עם 2 התחנות
---	---

<code>void addStation(Station newStation)</code>	פעולה המקבלת תחנה ומוסיפה אותה בסוף מסלול האוטובוס הקיים. הנח שתחנה זו איה קיימת במסלול
<code>double routeLength()</code>	פעולה המחזירה את אורך מסלול האוטובוס כלומר את סכום המרחקים בין תחנה לתחנה

- א. כתוב את כותרת המחלקה ואת התכונות שלה. כתוב תיעוד לכל תכונה, לתכונה המחזיקה את רשימת התחנות קרא `route`
- ב. ממש את הפעולה הבונה של המחלקה `BusRoute`
- ג. ממש את הפעולה המוסיפה תחנה למסלול-אוטובוס
- ד. ממש את הפעולה המחשבת את אורך מסלול האוטובוס
- ה. צור בתוכנית הראשית במחלקה `Program` קטע תוכנית שייצור את מסלול האוטובוס המתואר בדוגמא בתחילת השאלה וידפיס את אורך המסלול. פורמט ההדפסה
`route length: 12.077894`

בהצלחה