**חלק יבש – תרגיל בית 3:**

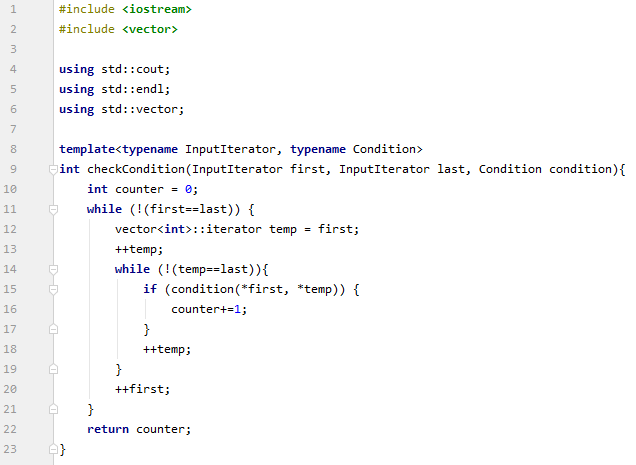
מגישים:

גיל לוין – 325814010

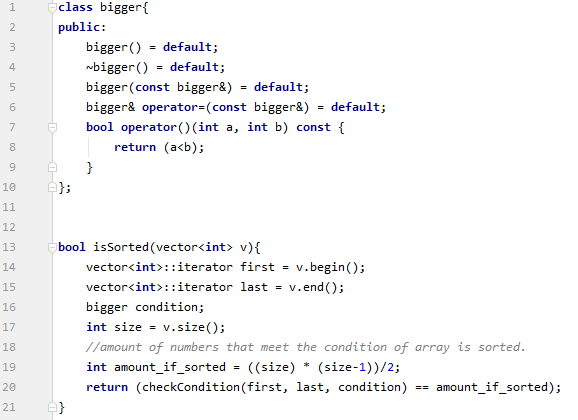
יהב בונה – 325755825

שאלה 1:

א)



ב) המשך קוד ל – א' (כלומר, כל הusing ושאר הדברים שהוגדרו ל – א' תקפים גם כאן):



הערה: bigger הוא פנקטור שמשמש כתנאי לcheckCondition.

שאלה 2:

א) מה שיודפס על ידי הקוד הוא:

“from Base”

“from A”

“from Base”

הסבר:

השורה הראשונה תודפס בגלל הקריאה לפונקציה baseMethod בשורה השנייה של main – הפונקציה מפעילה את הפונקציה method של Base, גם אם הclass יורש מBase, בגלל שbaseMethod לא virtual.

השורה השנייה תודפס בגלל שאנחנו עושים delete לbase, והזכרון שמוקצה עבור A אך לא עבור Base משוחרר (הזכרון הנוסף של class שיורש מclass הבסיס משוחרר לפני הזכרון שמוקצה עבור החלק של class הבסיס), כלומר נקרא הd’tor של A, וכתוצאה מכך מופעלת הפונקציה method של A (method היא virtual ולכן הפונקציות method של Base ושל A שונות), שמדפיסה “from A”.

השורה השלישית מודפסת בגלל שלאחר שמשוחרר הזכרון שמוקצה במיוחד עבור A, צריך לשחרר את הזכרון שמוגדר עבור Base, ולכן נקרא הd’tor של Base, שמפעיל את method של Base, שמדפיס “from Base”.

ב) על מנת לשנות את השורה הראשונה יש כמה אפשרויות: לשנות את הפונקציה baseMethod, כך שהיא תדפיס משהו אחר (ולא תקרא לmethod), אפשר לשנות את method של Base כך שהיא תדפיס הודעה אחרת, ואז כשbaseMethod תקרא לmethod, תודפס הודעה אחרת, ואפשר לשנות את הפונקציה baseMethod לvirtual ולהוסיף פונקציית baseMethod לA שמדפיסה הודעה אחרת מזו שמודפסת כרגע (“from Base”), כך שכשתיקרא הפונקציה baseMethod, תופעל הפונקציה baseMethod, של A.

ג) אפשר לעשות שני דברים: אפשר לכתוב בנאים לBase ולA, כך שכל בנאי ידפיס “from X” המתאים באמצעות קריאה לmethod, ואז בשורת האתחול יקרא הבנאי של Base, ולאחר מכן הבנאי של A וידפסו ההודעות המתאימות, ומאחר ואנחנו לא יודעים אם קריאה לmethod נחשבת כלכתוב שורת הדפסה, אז יש עוד אופציה שהיא למחוק את ה"~" מהd’tor-ים וכך להישאר עם בנאים שיעשו בדיוק מה שתואר באופציה הראשונה שהצגתי בסעיף זה.