

תכנות מונחה עצמים ושפת C++

300100

*

עבודת בית מחליפה את המבחן

מרצה: ד"ר רינה צביאל-גירשין
מתרגלת: מעיין זנו

תוכן עיניינים

- 2 הוראות כלליות:
- 3 הנחיות להגשה סופית של הפרויקט:
- 4 שאלה
- 7 אילוצי מימוש:

הוראות כלליות:

- משך העבודה: 72 שעות.
- **גרסת העבודה:** העבודה תתבצע בזוגות בהם עבדתם במהלך הקורס. שימו לב שיכול להיות שבני הזוג קיבלו עבודות שונות לביצוע. לכן, יש לבצע את העבודה שקיבל בן הזוג ששתי הספרות האחרונות של ת"ז שלו יוצרות מספר גדול יותר (למשל, אם שתי הספרות האחרונות של ת"ז של אחד מבני הזוג הן 18 ושל השני הן 45 – יש לבצע את העבודה של בן הזוג שת"ז שלו מסתיימת ב-45).
- **המלצה:** לא לעשות כל החלקים יחד. לחלק את העבודה בין בני הזוג.
- הקוד עבור כל המחלקות חייב להיות מלווה בהערות המסבירות את כל עקרונות המימוש. בתחילת כל מחלקה יש לכתוב הערה הכוללת הסבר על מהות המחלקה.
בנוסף יש לכתוב הערה בתחילת כל מתודה בה עליכם לפרט את מטרת המתודה וכן יש לתעד את השדות של כל מחלקה.
- את ההערות יש לכתוב בקבצי C++ ו H. אחד הקריטריונים לאיכות העבודה הוא היותה Well-Commented!!! חלק מן הנקודות של הציון יינתן עבור הערות שכתבתם.
- הקוד שתכתבו יישפט, בין היתר, על פי מידת עמידתו בעקרונות OOP שנלמדו ובעקרונות תוכנות כגון האורך והאלגנטיות שלו. מתודות או תוכניות ארוכות או מסורבלות ללא צורך יקבלו פחות נקודות, אפילו אם הן ממלאות את המשימה שהוגדרה בשאלה. יש להשתמש בשמות המחלקות ומשתנים בעלי משמעות!
- הקוד הסופי **חייב להתקמפל ולרוץ!** זהו תנאי סף לציון עובר.
- 15% מציון העבודה יינתנו עבור רמת השקעה ויצירתיות.
- **הגשות ביניים לאתר הקורס:**
- **בכל יום, בשעה 10:00** יש להגיש לתיבת ההגשה המתאימה באתר הקורס קובץ מכוון ZIP של תיקיית הפרויקט המכילה את מה שהספקתם לעשות עד לרגע זה בפרויקט.
יהיו 2 הגשות כאלה – 24 שעות לאחר פרסום העבודה ו- 48 שעות לאחר פרסום העבודה.
הקבצים שתגישו יבדקו בקפדנות וחלק מהניקוד על העבודה יהיה על הגשות אלה.
לא נקבל מצב שבו בסוף היום הראשון לא הוגש כלום או קוד כמעט ריק ולמחרת הוגש פרויקט כמעט שלם.
למען הסר ספק – הקוד שמוגש בהגשות הביניים לא חייב להתקמפל.
הלינקים לתיבות ההגשה נמצאים באתר הקורס.
- **מענה לשאלות במהלך ביצוע העבודה:**
המענה לשאלות בנוגע לעבודה יינתן דרך קובץ Excel שיתופי שהלינק אליו נמצא באתר הקורס. מענה לשאלות המופיעות בקובץ יינתן פעמיים ביום – בשעות הצהריים ובערב.
יש לקרוא את השאלות והתשובות להן כבר ניתן מענה, פן השאלה כבר נשאלה ע"י סטודנטים אחרים.
לא ניתן לפנות לסגל הקורס באמצעות דוא"ל/טלפון/וואטסאפ עם שאלות הנוגעות לעבודה.

הנחיות לאופן הגשת העבודה בעמוד הבא!

הנחיות להגשה סופית של הפרויקט:

תאריך ההגשה : **24/2/2022 בשעה 10:00** (רק אחד מבני הזוג מגיש את הקובץ לתיבה של בן הזוג ששתי הספרות האחרונות של ת"ז שלו יוצרות מספר גדול יותר)

אופן ההגשה:

- יש להגיש את המטלה בזוגות בלבד
- יש לרכז את כל הקבצים הנדרשים לפרויקט בקובץ מכוון אחד ZIP (או . RAR .
- שם הקובץ המכוון יקרא כך OOP_Name1_Studentid_Name2_Studentid כאשר
 - Name1 מציין את השם המלא (שם פרטי ושם משפחה) של סטודנט 1
 - Name2 מציין את השם המלא של סטודנט 2
 - שם הקובץ ייכתב באנגלית בלבד!
- עליכם לצרף לתיקייה המכוננת את כל התיקייה של הפרויקט הכוללת את כל קבצי הקוד של התוכנית (כולל קובץ SLN, כולל הערות וכולל קבצי נתונים).
- יש לממש הכל בפרויקט אחד.
- עליכם לצרף להגשה (לא בתוך ה-rar/zip מסעיף קודם אלא קובץ נוסף) קובץ WORD בשם OOP_Name1_Studentid_Name2_Studentid ובו:
 - בתחילת קובץ זה מופיעים שמות המבצעים, ת"ז, כתובות דוא"ל, **במודגש!**
 - העתק של כל הקוד מכל קובץ (כולל הערות)
 - עבור כל class שמועתק לשם, יש לציין את שם ה class הממומש
 - בנוסף, בסיום, יש לצרף דוגמת הרצה של התוכנית
 - צילום קבצי הטקסט בהם עשיתם שימוש או יצרתם
 - יש להעתיק את כל מה שכתבתם לקובץ WORD אחד.
- **אזהרה:** מטלות שתוגשנה לא בהתאם להוראות ההגשה לעיל, תספוגנה הורדה ניכרת בציון הסופי
- לא ניתן לאחר בהגשת הפרויקט הסופי! מעבר למועד האחרון להגשה (בשעה הנקובה!), המערכת תיסגר באופן אוטומטי להגשות ומי שלא יגיש עד לתאריך זה יקבל באופן אוטומטי ציון סופי 0 במשימה.
- נא לא לחכות לרגע האחרון כדי להגיש. מומלץ להתחיל את תהליך ההגשה לכל המאוחר שעה לפני סגירת המערכת.
- **הגנה בעל פה:** עבור חלק מן העבודות, באופן אקראי, צוות הקורס יבקש מסטודנטים לעמוד בהצלחה בהגנה בעל פה על עבודתם.

בהצלחה

שאלה

בעבודה זו בנוסף לקבצי הקוד וקבצי נתונים, עליכם להוסיף קובץ המכיל הסבר מפורט וברור לגבי המחלקות אותם בניתם בעבודה. ההסבר צריך להכיל תיאור של הקשרים בין המחלקות, תיאור המחלקה המופשטת והמתודות המדומות טהורות (כולל מדוע בחרתם בכל מתודה להיות מדומה טהורה), תיאור משמעות השדה הסטטי (חייבת להיות תכונה סטטית באחד המחלקות), וכן כל הסבר שבאמצעותו ניתן יהיה להבין את משמעות התכנון המחלקות שלכם.

עליכם לבחור בעצמכם נושא ולכתוב קבוצת מחלקות אשר באמצעותן תממשו נושא זה המתבססות על שקיבלתם. מומלץ להשקיע זמן ומחשבה רבה בתכנון הרעיון וחלוקת המחלקות. הקוד Person מחלקת בסיס צריך לתמוך בעקרון הפולימורפיזם.

נתונה מחלקת בסיס Person ומחלקה נוספת Address (הסבר בהמשך):

```
#ifndef __PERSON_H
#define __PERSON_H

#include <iostream>
using namespace std;

class Person //abstract
{
    char *name;
    long id;
    int age;
    double weight;

public:
    Person();
    Person(char* name, long id, int age, double weight);
    Person(const Person& other);
    ~Person();

    const Person& operator=(const Person& other);
    friend ostream& operator<<(ostream& out, const Person& p);

    bool operator==(Person& other);
    virtual void show() const = 0;

    char* getName() const {return name; }
    long getID() const { return id; }
    int getAge() const { return age; }
    double getWeight() const { return weight; }
    void setName(char* name);
    void setWeight(double weight);
    void setAge(int age);

    bool fat(); //over 100
    bool old(); //over 70
    bool healthy(); //true fat & true old
};

#endif
```

```

#ifndef _ Address_H_
#define _ Address_H_

#include <iostream>
#include <string>
using namespace std;

class Address
{
    char* city;
    string street;
    int houseNo;

public:
    Address(char* city, string street, int houseNo);
    Address();
    Address (const Address & other);
    ~Address ();
    const Address & operator=(const Address & other);
    virtual void show() const = 0;

    void setCity(char* city);
    void setStreet(string street);
    void setHouse(int h);

    friend ostream& operator<<(ostream &out, const Address & p);
    char* getCity() const;
    string getStreet () const;
    int getHouse() const ;
    bool operator== (const Date& otherDate) const;
};
#endif
  
```

1. ממחלקה Person יש לגזור לפחות שתי מחלקות לפי בחירתך. לכל מחלקה יש להוסיף לפחות 2 שדות, 3 בנאים, מתודות set,get לבחירתכם (עבור כל שדה החליטו אם תרצו לאפשר שיראו או יעדכנו אותו), חלק ממימושי ה-set צריכים לבדוק תקינות נתונים ולזרוק חריגות במקרה שהנתונים לא תקינים, בנאי העתקה, אופרטור =, מחסל, לפי הצורך, אופרטור == (שימו לב, שנרצה שתעשו דריסה לאופרטור זה ולא העמסה), מתודת הדפסה show, אופרטור <<, שתי מתודות נוספות לבחירתך ושני אופרטורים נוספים לבחירתך.
2. מאחד מהבנים של Person יש לגזור צאצא (נכד של Person). למחלקת צאצא יש להוסיף לפחות 2 שדות כאשר אחד השדות צריך להיות מסוג תור Queue. בנוסף יש להוסיף לפחות 2 בנאים, תכונות שזורקות חריגות במקרה שהנתונים לא תקינים, שתי מתודות נוספות, מתודת הוספת פריט לתור Add, אופרטור ==, אופרטור <<, ומתודת הדפסה show.

3. דרישות נוספות :

- באחת המחלקות חוץ ממחלקת בסיס שקיבלתם מאיתנו יהיה שדה מסוג מצביע
- במקרה הצורך מותר להוסיף למחלקות שקיבלתם מתודות ואופרטורים (יש לנמק מה הוספתם ומדוע)
- יש לממש קשרי הורשה והכלה
 - לאחת המחלקות יש להוסיף שדה מסוג אובייקט מוכל Address
- לפחות אחת המחלקות צריכה להיות מחלקה מופשטת והיא תכיל לפחות מתודה מדומה טהורה אחת (יש לפרט בקובץ WORD)
- לפחות אחת מהמחלקות תכיל שדה סטטי (יש לפרט בקובץ WORD)
- לפחות אחת מהמחלקות תכיל מתודה או פונקציה חברה (אין חובה לממש מחלקה חברה אולם אפשרי להוסיף זאת בנוסף לדרישה זו במידה ותצטרכו).
- לפחות במחלקה אחת יהיה שדה מסוג מיכל/מבנה (מתוך STL) ויש לספק מתודות שישתמשו באיטרטור iterator בשביל לעבור על תוכן המיכל

4. יש לכתוב מחלקה שמנהלת את התוכנית מחלקה MenuManager מאגדת את המחלקות הקודמות למחלקה אחת. במחלקה זו השדות יהיו אובייקטים מסוג המחלקות שמימשתם בסעיפים 1 ו-2. (לדוגמה : מחלקה כיתה בה לומדים אנשים שונים שפה זרה. אסור להשתמש בדוגמה זו). יש לבחור את המיכל/מבנה נכון (מתוך STL) לשמירת האובייקטים.. יש לממש 3 בנאים, מתודת הוספת ומחיקת איבר, תכונות שזורקות חריגות במקרה שהנתונים לא תקינים, מתודת חיפוש איבר (יש להחליט לפי מה מחפשים. יש להסביר את ההחלטה בקובץ WORD), אופרטור <<, ומתודת הדפסה. במחלקה זו תגדירו מתודות שונות בהם יבוא לידי ביטוי מנגנון הפולימורפיזם (חשבו כיצד משפיע רעיון זה על שאר המחלקות) ותפעילו את המתודות והאופרטורים אותם מימשתם במחלקות השונות לפי ראות עיניכם. בנוסף יש לממש מתודה StartProgram אשר תדפיס למשתמש מהו נושא התוכנית והאופציות השונות. המשתמש יוכל לבחור בכל פעם אופציה מבין האופציות בתפריט עד שיחליט לסיים. יש לבצע ניהול שלם של התוכנית דרך מתודה הזאת הפעילו מימושים של מתודות/אופרטורים אחרים שקיימים במחלקה. יש לתפוס חריגות, לבחירתך כיצד להתנהל במצב של תפיסת חריגה בהתאם לסיפור (רשמו בקובץ WORD הסבר מדוע בחרת באופן זה לטפל בחריגות).

אילוצי מימוש:

- יש ליצור לפחות 4 מחלקות (בין חלקן צריך להיות יחס הורשה ואובייקטים מוכלים)
- למחלקה אחת חייבים להיות ריבוי אבות . בכל אחד מן האבות חייב להיות שדה name , אופרטור == , אופרטור < , ומתודת הדפסה show. אחד האבות יכולה להיות מחלקה שלא קשורה למחלקת בסיס שקיבלתם. (יש לפרט בקובץ WORD)
- יש ליצור קובץ נתונים ראשוני לכל מחלקה לא אבסטרקטית, בקובץ זה יהיו רשומים לפחות 5 אובייקטים וכאשר התוכנית תעלה בפעם הראשונה יש לשמור את הנתונים הללו בתוכנית וכאשר המשתמש בוחר לסיים את התוכנית יש לשמור כל הנתונים לקובץ בחזרה.
- יש להשתמש בשני סוגי מיכלים /מבנים שונים (אחד כשדה באחת המחלקות) ושני בפונקציה עיקרית.
- יש לזרוק לפחות 3 חריגות שונות (יש לפרט בקובץ WORD)
- יש לכתוב פונקציה עיקרית main :
- יש להגדיר אובייקט מסוג המחלקה שהגדרת בסעיף 4. עליכם לאפשר להריץ קוד בצורה אוטומטית (קוד זה יכלול יצירת אובייקטים מקבצים, יוצר אובייקטים מכל המחלקות שמימשתם בסעיפים 1,2. יש ליצור לפחות 2 אובייקטים מכל סוג, להשים/להוסיף אותם למבנה, להפעיל את המתודה StartProgram אותה ממשותם במחלקה בסעיף 4 אשר מנהלת את כל הפעולות השונות לפי הסיפור שבחרתם.
- בסוף העבודה יש ליצור קובץ אחד OUT.TXT שבו ירשמו כל האובייקטים (מסוג סעיף 1 ו2) שקיימים במיכל או מחלקה מאגדת לאחר כל השינויים שנעשו איתם. פורמט ההדפסה יהיה – שם המחלקה – נתוני האובייקט. שם המחלקה יש לרשום באמצעות typeid