**תרגיל 9**

**STL**

בתרגיל זה נחזור וננהל את המוסד להשכלה גבוהה שעסקנו בו בתרגיל 7, תוך שימוש במחלקות ובאלגוריתמים של ספריות STL.

לשם כך הבא מתרגיל 7 את המחלקות הבאות: Student, BA, MA, PHD.

1. הוסף לכל אחת מהמחלקות הנ"ל פונקציה בשם string studType() שתפקידה להחזיר מחרוזת שמציינת באיזה טיפוס סטודנט אובייקט זה מייצג.

(הפונקציה תחזיר "BA","MA","PHD" בהתאם לטיפוסי הסטודנט השונים)

1. בתוכנית הראשית נגדיר ווקטור vector או רשימה list (לבחירתך) בעזרת STL container שיכול להכיל סטודנטים מ**כל** הסוגים (פירוט בהמשך) אך תחילה עליך להגדיר מספר פונקציות גלובליות שפועלות על ווקטור או רשימה של סטודנטים (מכל הסוגים) לפי הפירוט הבא:

* הגדר פונקציה בשם add שמקבלת רשימה או וקטור (של STL) ומוסיפה לתוכו סטודנט חדש .

לשם מימוש הפונקציה יש להשתמש בפלטים הבאים בלבד:

cout<<"enter 1 to add a BA student\n";

cout<<"enter 2 to add a MA student\n";

cout<<"enter 3 to add a PHD student\n";

cout<<"enter id, first name, last name and number Of Courses\n";

cout<<"enter "<<numOfCourses<<" grades\n";

cout<<"enter 1 if the student does research and 0 if not\n";

cout<<"enter number of research hours\n";

* (חזרה על תרגיל 7 רק עם שימוש ב – stl):

הגדר פונקציה גלובלית בשם milga שמקבלת רשימה או ווקטור של סטודנטים ומדפיסה מתוכה רק את הפרטים של הסטודנטים הזכאים למלגה .

* הגדר פונקציה גלובלית בשם mostResearchHours שמקבלת רשימה או ווקטור של סטודנטים ומחזירה מצביע לסטודנט ברשימה שלומד לתואר שלישי (PHD) ומשקיע הכי הרבה שעות במחקר. במידה ואין סטודנטים לתואר שלישי בכלל הפונקציה תחזיר NULL

1. נתונה בהמשך התוכנית הראשית,

בתוכנית זו נרצה להגדיר ווקטור vector או רשימה list (לבחירתך) בעזרת STL container שיכול להכיל סטודנטים מ**כל** הסוגים (BA,MA,PHD). ולאחר מכן לאפשר למשתמש לבחור (בלולאה) פעולה לביצוע עד לבחירת STOP.

הפעולות האפשריות:

0 – סיום התוכנית

1 – הוספת סטודנט חדש.

2 – הדפסת נתוני הסטודנטים שזכאים למלגה

3 - הדפסת שם (משפחה ופרטי) של הסטודנט לתואר שלישי שעוסק הכי הרבה במחקר

4 – הדפסת מספר הסטודנטים לתואר ראשון שזכאים למלגה

5 - הדפסת שמות (משפחה ופרטי) הסטודנטים לתואר שני שאינם עוסקים במחקר.

6 – הדפסת הודעה האם קיים במערכת סטודנט כלשהו שלומד יותר מ- 15 קורסים.

7 – מחיקה מהרשימה/וקטור של כל הסטודנטים לתואר שלישי שאינם לומדים קורסים בכלל.

עליך להשלים את שורות הקוד החסרות,

שים לב – בכל מקום, יש להשלים שורה אחת בלבד.

(רמז: העזר בביטויי lambda וקרא לפונקציות האלגוריתמיות שמוגדרות בספריה algorithm של STL.

#include <iostream>

#include <list>

#include <vector>

#include <algorithm>

#include <string>

using namespace std;

enum option{

|  |  |  |
| --- | --- | --- |
| stop, | // | סיום התוכנית |
| addNewStudent, | // | הוספת סטודנט חדש |
| milgaList, | // | הדפסת פרטי כל התלמידים הזכאים למלגה |
| mostResearch, | // | הדפסת שם (משפחה ופרטי) של הסטודנט שעוסק הכי הרבה שעות במחקר |
| countMilgaBA, | // | הדפסת מספר הסטודנטים לתואר ראשון הזכאים למלגה |
| noResearchMA, | // | הדפסת שמות (משפחה ופרטי) של הסטודנטים MA שאינם עוסקים במחקר |
| overheadStudent, | // | הדפסת הודעה מתאימה, האם קיים סטודנט שלומד יותר מ- 15 קורסים |
| removePHDFailers, | // | מחיקת כל הסטודנטים לתואר שלישי שאינם לומדים באף קורס |

};

void add(\_\_\_\_\_ווקטור או רשימה\_\_\_\_)

void milga (\_\_\_\_\_ווקטור או רשימה\_\_\_\_)

Student\* mostResearchHours (\_\_\_\_\_ווקטור או רשימה\_\_\_\_)

int main()

{

// הצהרה על ווקטור או רשימה של סטודנטים

int op;

cout<<"enter 0-7\n";

cin>>op;

while(op!=stop)

{

switch (op)

{

case addNewStudent:add(\_\_\_\_\_\_\_\_\_\_\_\_);//הוספת סטודנט חדש break;

case milgaList:milga(\_\_\_\_\_\_\_\_\_\_\_\_); //הדפסת פרטי הזכאים למלגה

break;

case mostResearch: {

//הדפסת שם הסטודנט שעוסק במחקר הכי הרבה שעות:

Student\* s=mostResearchHours(\_\_\_\_\_\_\_\_\_\_\_\_);

cout<<"PHD studnet with most research hours: ";

cout <<s->getFname()<<' '<<s->getFname()<<endl;

}

break;

case countMilgaBA: cout<<"#BA studnets for milga: ";

//הדפסת מספר הזכאים למלגה בתואר הראשון

cout<<endl;

break;

case noResearchMA:cout<<"list of no research MA students : ";

//הדפסת רשימת (שם משפחה ופרטי) הסטודנטים לתואר שני שאינם עוסקים במחקר

cout<<endl;

break;

case overheadStudent:

if(קיים תלמיד שלומד יותר מ- 15 קורסים)

cout<<"there is a student that takes more than 15 courses\n";

else cout<<"no student takes more than 15 courses\n";

break;

case removePHDFailers:

//מחיקה מהווקטור/רשימה של כל הסטודנטים לתואר שלישי שאינם לומדים קורסים בכלל

//הדפסת כל הרשימה אחרי מחיקת האיברים

break;

};

cout<<"enter 0-7\n";

cin>>op;

}

return 0;

} }