



אוניברסיטת אריאל בשומרון

פקולטה: מדעי הטבע

מחלקה: מדעי המחשב

שם הקורס: מבנה זיכרון ושפת C++

קוד הקורס: 2-7027810 כל הקבוצות - קבוצות 1,3,4,5

מועד א סמסטר: ב שנה: ה'תשעט

תאריך הבחינה: ט"ו בסיון ה'תשע"ט, 18/06/2019

משך הבחינה: 3 שעות

שם המרצים: אראל סגל-הלוי, גיל בן-ארצי

- יש לענות על כל השאלות במחברת הבחינה.
- אין צורך להעתיק את השאלון למחברת - השאלון יתפרסם בגיטהאב לאחר הבחינה.
- אסור להשתמש בכל חומר עזר.
- בשאלות הבנת קוד וזיהוי שגיאות, יש להסביר בפירוט מה גורם לשגיאה.
- בשאלות התיכנות, יש לכתוב קוד נכון ומסודר לפי כללי התיכנות שנלמדו בהרצאות ובתרגולים.
- בשאלות התיכנות, יש לכתוב הערות מפורטות בעברית או באנגלית, המסבירות את אופן הפתרון.

בהצלחה!!!

שאלה 1 [9 נק']

```
1. #include <iostream>
2. using namespace std;
3. class A {
4. public:
5.     A() { cout << "A(0)" << endl; }
6.     A(const A& a) { cout << "A(1)" << endl; }
7. };

8. class B {
9. public:
10.     B() : a() { cout << "B(0)" << endl; }
11.     B(const B& b) { cout << "B(1)" << endl; }
12. private:
13.     A a;
14. };
15.
16. int main() {
17.     B object1;
18.     B object2 = object1;
19.     return 0;
20. }
```

א [3 נק']. מה פלט התוכנית?

ב [3 נק']. אם מוחקים את שורה 11, מה פלט התוכנית?

ג [3 נק']. אם מחזירים את שורה 11 ומוחקים את שורה 6, מה פלט התוכנית?

שאלה 2 [9 נק']

```
1. #include <iostream>
2. using namespace std;
3.
4.     class A;
5.
6.     class B {
7.     public:
8.         explicit B (A& o) { cout << "B (A)" << endl; }
9.     };
10.
11.    class A {
12.    public:
13.        A() { cout << "A()" << endl; }
14.    };
15.
16.    void printit(B arg) {}
17.
18.    int main() {
19.        A a;
20.        B b(a);
21.        printit(a);
22.        return 0;
23. }
```

א [3 נק']. בתוכנית יש שגיאת קומפילציה בשורה 21. הסבירו במילים שלכם מהי השגיאה ומדוע היא קורה.

ב [3 נק']. הציעו תיקון לשורה 21 כך שהתוכנית תתקמפל ותרוץ בלי שגיאות. אין לשנות שורות אחרות.

ג [3 נק']. מהו פלט התוכנית לאחר התיקון?

שאלה 3 [9 נק']

```
1. #include <iostream>
2.
3. using namespace std;
4.
5. class Shape {
6. public:
7.     virtual void draw() = 0;
8.     virtual ~Shape() { cout << "~Shape()" << endl; }
9. };
10.
11.
12. class Circle : public Shape {
13. public:
14.     void draw() { cout << "Circle::draw()" << endl; }
15.     ~Circle() { cout << "~Circle()" << endl; }
16. };
17.
18.
19. void drawTheShape(Shape s) {
20.     s.draw();
21. }
22.
23. int main() {
24.     Circle c;
25.     drawTheShape(c);
26.     return 0;
27. }
```

א [3 נק']. בתוכנית יש שגיאת קומפילציה בשורה 19. הסבירו במילים שלכם מהי השגיאה ומדוע היא קורה.

ב [3 נק']. הציעו תיקון לשורה 19 כך שהתוכנית תתקמפל ותרוץ בלי שגיאות. אין לשנות שורות אחרות.

ג [3 נק']. מהו פלט התוכנית לאחר התיקון?

שאלה 4 [9 נק']

נתונה התוכנית:

```
01  #include <iostream>
02  #include <string>
03  using namespace std;
04
05  class string {
06      const char* p;
07  public:
08      string(const char* p) {this->p = p;}
09  };
10
11  int main() { // a demo program
12      string s = "abc";
13  }
```

כשמריצים make מתקבלת ההודעה הבאה:

```
main.cpp:12:2: error: reference to 'string' is ambiguous
string s = "abc";
^
```

א [3 נק'] מה סוג השגיאה (קומפילציה / קישור / זמן-ריצה / לוגית / דליפת זיכרון)? מה בדיוק גורם לשגיאה?

ב [3 נק'] איך אפשר לתקן את הבאג ע"י **מחיקה** של שורה אחת? אין לשנות את התוכנית הראשית.

ג [3 נק'] במחלקה הנתונה יש פוינטר p אבל אין מפרק (destructor). האם זה עלול לגרום לשגיאה כלשהי? אם כן, מה סוג השגיאה שעלולה להיווצר? אם לא, מדוע לא?

שאלה 5 [22 נק']

סטודנט כתב תבנית-מחלקה (class template) בשם Tree המייצגת עץ-מיון בינארי.

כיתבו בדיקות-יחידה מקיפות עבור המחלקה. יש לבדוק את הדרישות הבאות:

- אפשר ליצור עץ בינארי עם כל סוג שיש לו אופרטור "קטן מ-" (>).
- העץ אמור לשמור על סדר הפריטים בתוכו, בהתאם לאופרטור "קטן מ-" שלהם.
- העץ אמור לאפשר הכנסה (insert), וכן לבדוק מי הבן השמאלי (left) של פרט כלשהו בעץ.

א [10 נק']. תיכנון הבדיקות: כיתבו (בעברית) רשימה מפורטת של מקרים שאתם מתכוונים לבדוק.

ב [10 נק']. מימוש הבדיקות: מצורף קובץ עם בדיקות לדוגמה. עליכם להרחיב אותו ולהוסיף לו את הבדיקות בהתאם לתיכנון שכתבתם בסעיף א.

הפונקציות CHECK_OK, CHECK_EQUAL, CHECK_THROWS כבר מוגדרות בקובץ badkan.hpp, כמו במטלות. אין צורך לממש אותן.

מותר להוסיף, לצורך הבדיקות, מחלקות ו/או פונקציות לפני התוכנית הראשית.

```
#include <iostream>
#include "badkan.hpp"
#include "Tree.hpp"
using namespace std;

int main() {
    ariel::Tree<string> stree;
    ariel::Tree<int> itree;
    badkan::TestCase testcase("Binary tree");
    testcase
        .setname("Tree of string")
        .CHECK_OK (stree.insert("eee"))
        .CHECK_OK (stree.insert("ccc"))
        .CHECK_OK (stree.insert("ggg"))
        .CHECK_EQUAL (stree.left("eee"), string("ccc"))
            // the left child of "eee" is "ccc"
        .CHECK_THROWS(stree.left("ccc"))
            // "ccc" is a leaf - it has no left child
        .setname("Tree of int")
        .CHECK_OK (itree.insert(5))
        .CHECK_THROWS(itree.left(9)) // 9 does not exist in the tree
        // Add more tests here
        .print();
}
```

שאלה 6 [22 נק']

הגדירו תבנית (template) של מחלקה בשם `matrix` המייצגת מטריצה. יש לממש את ה `template` בשם כך שניתן יהיה להריץ בעזרתו את התוכנית הראשית ללא שגיאות. המחלקה צריכה לתמוך בפעולות הבאות:

- איתחול (בניה) מתוך וקטור של וקטורים. חתימת הבנאי:

```
Matrix(const std::vector<std::vector<T>>& V);
```

- הדפסת מטריצה בעזרת אופרטור הפלט;

- חיבור שתי מטריצות באותו גודל בעזרת אופרטור `+`;

- כפל מטריצה בוקטור בעזרת אופרטור `*`.

$$\begin{bmatrix} 1 & -1 & 2 \\ 0 & -3 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix}$$

הגדרת מכפלת מטריצה בוקטור: נתונה מטריצה $K \times M$ ווקטור $M \times 1$. המכפלה שלהם הינה מטריצה/וקטור בגודל $K \times 1$. האיבר ה- j הינו המכפלה של השורה ה- j במטריצה בוקטור הנתון.

$$\begin{bmatrix} 2 \cdot 1 - 1 \cdot 1 + 0 \cdot 2 \\ 2 \cdot 0 - 1 \cdot 3 + 0 \cdot 1 \end{bmatrix}$$

לדוגמא:

$$\begin{bmatrix} 1 \\ -3 \end{bmatrix}$$

הערות:

- בנוסף לפונקציות הכתובות למעלה, יש לממש כל פונקציה נוספת הדרושה כדי שהתוכנית תעבוד.
- בכל מצב של קלט לא תקין, יש לזרוק חריגה.

```
int main() {
    std::vector<std::vector<int>> Mat {{1,2,3},{4,5,6},{7,8,9}};
    Matrix<int> a(Mat); // constructs a matrix from vector of vectors
    Matrix<int> b ( {
        {4, 5, 6},
        {9, 0, 1},
        {7, 8, 7} });

    std::cout << "a: \n" << a;
    std::cout << "b: \n" << b;
    std::cout << "a + b: \n" << a+b; // should print:
        // 5 7 9
        // 13 5 7
        // 14 16 16
    Matrix<int> d ( {
        {1, 2},
        {4, 5},
        {7, 8} });

    std::vector<int> d_vector = {1, 2};
    std::cout << "d * d_vector: \n" << d * d_vector; // should print:
        // 5
        // 14
        // 23

    return 0;
}
```

שאלה 7 [22 נק']

כיתבו מחלקה בשם Money, המייצגת סכום-כסף מסויים במטבע מסויים. המחלקה צריכה לאפשר חיבור של סכומי-כסף במטבעות שונים, תוך המרה אוטומטית של יחידות.

שער-החליפין של כל מטבע, בשקלים, נקבע בתחילת התוכנית.

מצורפת תוכנית דוגמה. עליכם לממש את כל הפונקציות הדרושות כדי שהתוכנית תעבוד בהתאם להערות המופיעות בגוף הקוד. אין צורך לממש פונקציות נוספות. הערות:

- אין צורך להתייחס לשגיאות עיגול של מספרים ממשיים בפלט.
- עליכם לכתוב שני קבצים: Money.hpp ו-Money.cpp.

```
#include <iostream>
using namespace std;
#include "Money.hpp"
using ariel::Money;

int main() {
    // Set the exchange rates in shekels:
    Money::set_rate("ILS", 1);
    Money::set_rate("USD", 4);
    Money::set_rate("EUR", 6);

    // Define amounts and do calculations:
    Money shekel1(1, "ILS");
    Money dollar3(3, "USD");
    Money sum = shekel1 + dollar3;
    cout << sum << endl; // 13 ILS
    cout << (dollar3 + Money(1, "EUR")) << endl; // 4.5 USD
    (sum += Money(2, "EUR")) += Money(0.5, "ILS");
    cout << sum << endl; // 25.5 ILS
    // Money(1, "XXX"); // throws an exception
}
```


שאלה 8 - מענקים

א. האם השתמשנו בבדיקות שלכם באחת המטלות? אם כן אנא ציינו באיזו מטלה בדיוק.

ב. האם זכיתם באחד משלושת המקומות הראשונים בתחרות בול-פגיעה? אם כן ציינו באיזה מקום.