



אוניברסיטת אריאל בשומרון

פקולטה: מדעי הטבע

מחלקה: מדעי המחשב

שם הקורס: מבנה זיכרון ושפת C++

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

מועד א___ סמסטר ב___ תאריך בחינה: 4/7/2018

משך הבחינה: 3 שעות

שם המרצים: אראל סגל-הלוי, חרות סטרמן

יש לענות על כל השאלות במחברת הבחינה.

אין צורך להעתיק את השאלון למחברת - השאלון יתפרסם במודל לאחר הבחינה.

בשאלות 1-5, כששואלים "מה סוג השגיאה", יש לכתוב אחת מהאפשרויות הבאות:

- שגיאת קומפילציה (compilation error)
- שגיאת קישור (link error)
- שגיאת זמן-ריצה (runtime error)
- שגיאה לוגית (logic error)

כששואלים "מה גורם לשגיאה" יש לפרט ולציין מספרי שורות בהתאם לקוד המובא בשאלה.

כששואלים "איך לתקן את השגיאה" יש לציין מספרי שורות ולהסביר מה צריך לשנות בכל שורה/שורות.

בחלק מהמקרים ישנם כמה תיקונים אפשריים. ניקוד מלא יינתן רק לתיקון הטוב ביותר מנקודת מבט של הנדסת תוכנה, כפי שנלמד בכיתה.

יש לענות תשובות מלאות אך ממוקדות. לא יינתנו נקודות על תשובות עם טקסט מיותר שאינו קשור לנושא.

ניתן לענות על כל שאלה במילים "לא יודע" או "לא יודעת". במקרה זה השאלה לא תיבדק כלל, ואתם תקבלו 20% מהניקוד על אותה שאלה (מעוגל כלפי מטה).

אסור להשתמש בכל חומר עזר.

בהצלחה!!!

שאלה 1 [10 נק']

נתונה התוכנית:

```
01  #include <iostream>
02  using namespace std;
03
04  class triplet {
05      double x,y,z;
06  public:
07      triplet(double x, double y,double z): x(x), y(y), z(z) {}
08      friend ostream& operator<< (ostream& out, const triplet& t);
09  };
10
11  ostream& operator<< (ostream& out, const triplet& t) {
12      out << "(" << t.x << "," << t.y << "," << t.z << ")";
13  }
14
15  int main() {
16      triplet t(1,2,3);
17      cout << t << endl;
18  }
```

כשמריצים make, מתקבל הפלט הבא:

```
main.cpp:13:1: warning: control reaches end of non-void function [-Wreturn-type]
}
^
1 warning generated.
./a.out
make: *** [all] Illegal instruction (core dumped)
```

א [5 נק'] מה סוג השגיאה (קומפילציה / קישור / זמן-ריצה / לוגית)? מה בדיוק גורם לשגיאה?

ב [5 נק'] יש לתקן את הבאג ע"י שינוי/הוספה של שורה אחת. איזה שינוי יש לבצע כך שהתוכנית תרוץ בלי שגיאות ותדפיס (1,2,3)?

שאלה 2 [10 נק']

נתון הקוד החלקי הבא (שלוש נקודות מציינות קוד שהושמט כי אינו רלבנטי לשאלה):

```
01  #include <iostream>
02  using namespace std;
03
04  class LargeMatrix {
05  public: // ...
06      LargeMatrix(int width, int height) {
07          cout << "standard ctor" << endl; // ...
08      }
09      LargeMatrix(const LargeMatrix& other) {
10          cout << "copy ctor" << endl; // ...
11      }
12  };
13
14  int sum_of_values(LargeMatrix m) {
15      cout << "sum_of_values" << endl;
16      int result = 0;
17      // do calculation ...
18      return result;
19  }
20
21  int main() {
22      LargeMatrix m1(1000,1000);
23      // ...
24      cout << sum_of_values(m1) << endl;
25  }
```

התוכנית מתקמפלת ורצה, אבל יש בה באג הגורם לכך שהתוכנית רצה מאד לאט.
הפלט המתקבל כולל את ארבע השורות הבאות:

```
standard ctor
copy ctor
sum_of_values
0
```

א [5 נק'] מהו הבאג? (ציינו מספרי שורות).

ב [5 נק'] יש לתקן את הבאג ע"י שינוי/הוספה של שורה אחת. איזה שינוי יש לבצע כך שהתוכנית תרוץ מהר ובלי שגיאות?

שאלה 3 [10 נק']

נתונה התוכנית הבאה:

```
class XX {
    int ixx;
};

class YY: public XX {
    int iyy;
};

class ZZ {
    YY oyy;
    int izz;
};

int main() {
    ZZ ozz;
}
```

התוכנית בשפת C++. התוכנית מתקמפלת ורצה בלי שגיאות.

א [2 נק']. באיזה איזור בזיכרון נמצא המשתנה `ixx` (מחסנית / ערימה / אחר)?

ב [2 נק']. באיזה איזור בזיכרון נמצא המשתנה `iyy` (מחסנית / ערימה / אחר)?

ג [2 נק']. באיזה איזור בזיכרון נמצא המשתנה `izz` (מחסנית / ערימה / אחר)?

ד [2 נק']. התוכנית תורגמה לשפת Java. התוכנית הראשית נשארה ללא שינוי:

```
public static void main(String[] args) {
    ZZ ozz;
}
```

התוכנית מתקמפלת ורצה בלי שגיאות.

באיזה איזור בזיכרון נמצא המשתנה `ixx` (מחסנית / ערימה / אחר)?

ה [2 נק']. התוכנית תורגמה לשפת Java. התוכנית הראשית שונתה ל:

```
public static void main(String[] args) {
    ZZ ozz = new ZZ();
}
```

התוכנית מתקמפלת ורצה בלי שגיאות.

באיזה איזור בזיכרון נמצא המשתנה `ixx` (מחסנית / ערימה / אחר)?

שאלה 4 [10 נק']

נתונה תוכנית המורכבת משלושה קבצים.

AbstractShape.h:

```
01
02 class AbstractShape {
03 public:
04     virtual void plot();
05 };
```

Circle.h:

```
11 #include "AbstractShape.h"
12 #include <iostream>
13 class Circle: public AbstractShape {
14 public:
15     void plot() { /* code for plotting a circle. not shown. */ }
16 };
```

main.cpp:

```
21 #include "Circle.h"
22 int main() {
23     AbstractShape* s = new Circle();
24     s->plot();
25 }
```

כשמריצים make מתקבלת הודעת השגיאה הבאה:

```
/tmp/main-3bcff2.o: In function `AbstractShape::AbstractShape()':
main.cpp:(.text._ZN13AbstractShapeC2Ev[_ZN13AbstractShapeC2Ev]+0x6):
undefined reference to `vtable for AbstractShape'
```

א [5 נק'] מה סוג השגיאה (קומפילציה / קישור / זמן-ריצה / לוגית)? מה בדיוק גורם לשגיאה?

ב [5 נק'] יש לתקן את הבאג ע"י שינוי/הוספה של שורה אחת. איזה שינוי יש לבצע כך שהתוכנית תרוץ בלי שגיאות ותצייר עיגול?

שאלה 5 [10 נק']

נתונה התוכנית:

```
01  #include <set>
02  #include <iostream>
03  using namespace std;
04
05  struct XX {
06      int x;
07      XX(int x): x(x) {}
08  };
09
10  int main() { // the program should print "100"
11      multiset<XX> myset;
12      myset.emplace(1);
13      myset.emplace(0);
14      myset.emplace(0);
15      for (auto y: myset)
16          cout << y.x;
17  }
```

כשמריצים make מתקבל הפלט הבא:

```
/usr/bin/.../lib/gcc/x86_64-linux-gnu/7.1.0/.../.../.../include/c+
+/7.1.0/bits/stl_function.h:386:20: error: invalid operands to
binary expression ('const XX' and 'const XX')
```

```
{ return __x < __y; }
```

```
~~~ ^ ~~~
```

```
/usr/bin/.../lib/gcc/x86_64-linux-gnu/7.1.0/.../.../.../include/c+
+/7.1.0/bits/stl_tree.h:2069:10: note: in instantiation of member
function 'std::less<XX>::operator()' requested here
```

```
__x = _M_impl._M_key_compare(__k, _S_key(__x)) ?
```

```
^
```

```
/usr/bin/.../lib/gcc/x86_64-linux-gnu/7.1.0/.../.../.../include/c+
+/7.1.0/bits/stl_tree.h:2381:19: note: in instantiation of member
function 'std::_Rb_tree<XX, XX, std::_Identity<XX>, std::less<XX>,
std::allocator<XX> >::_M_get_insert_equal_pos' requested here
```

```
auto __res = _M_get_insert_equal_pos(_S_key(__z));
```

```
^
```

```
/usr/bin/.../lib/gcc/x86_64-linux-gnu/7.1.0/.../.../.../include/c+
+/7.1.0/bits/stl_multiset.h:448:16: note: in instantiation of
function template specialization 'std::_Rb_tree<XX, XX,
std::_Identity<XX>, std::less<XX>, std::allocator<XX>
>::_M_emplace_equal<int>' requested here
```

```
{ return _M_t._M_emplace_equal(std::forward<_Args>(__args)...); }
```

```
^
```

```
main:12:8: note: in instantiation of function template  
specialization 'std::multiset<XX, std::less<XX>, std::allocator<XX>  
>::emplace<int>' requested here  
myset.emplace(1);
```

א [5 נק']. מה סוג השגיאה (קומפילציה / קישור / זמן-ריצה / לוגית)? מה בדיוק גורם לשגיאה?

ב [5 נק']. איך אפשר לתקן את התוכנית כך שתרוץ בלי שגיאות ותדפיס 100?

שאלה 6 [40 נק']

נתונה התוכנית:

```
01  #include <cmath>
02  #include <iostream>
03  #include <vector>
04  using namespace std;
05
06  class PositiveMatrix {
07      double* vals;
08      int rows,cols;
09  public:
10      PositiveMatrix(int rows, int cols):
11          rows(rows), cols(cols), vals(new double[rows*cols]) {
12          for (int i=0; i<rows*cols; ++i)
13              vals[i]=0.0;
14      }
15
16      double operator() (int x, int y) const {
17          return vals[x*cols+y];
18      }
19
20      double& operator() (int x, int y) {
21          return vals[x*cols+y];
22      }
23  };
24
25  int main() {
26      PositiveMatrix m1(10,4);
27      cout << m1(1,2) << endl; // prints 0 (OK)
28      m1(1,2) = 3.0;
29      cout << m1(1,2) << endl; // prints 3 (OK)
30
31      PositiveMatrix m2{m1};
32      m2(1,2) = 5;
33      cout << m1(1,2) << endl; // prints 5 (should print 3)
34
35      m1 = m2;
36      m1(1,2) = 7;
37      cout << m2(1,2) << endl; // prints 7 (should print 5)
38
39      m1(1,2) = -5.0; // should throw an exception "illegal value"
40  }
```


התוכנית מתקמפלת ורצה אולם המחלקה PositiveMatrix לא גמורה - יש להשלים בה מספר חלקים לפי הפירוט בסעיפים הבאים. **אין לשנות את התוכנית הראשית** - עליכם לוודא שהתוכנית הראשית כפי שהיא כתובה עכשיו רצה בלי שגיאות. אם השינוי שלכם דורש להחליף קוד שכבר קיים במחלקה PositiveMatrix - ציינו בפירוט את מספרי השורות שבהן יש להחליף את הקוד ומה יש לכתוב במקומו.

א [10 נקודות]. יש להוסיף קוד למחלקה PositiveMatrix כך שלא תהיה דליפת זיכרון בתוכנית.

ב [10 נקודות]. יש להוסיף קוד למחלקה PositiveMatrix כך שבשורה 33 יודפס הערך הנכון - 3.

ג [10 נקודות]. יש להוסיף קוד למחלקה PositiveMatrix כך שבשורה 37 יודפס הערך הנכון - 5.

ד [10 נקודות]. יש להוסיף ולשנות קוד כך שכל ניסיון לשנות ערך במטריצה למספר שלילי (כמו בשורה 39) יגרום לזריקת חריגה.

שאלה 7 [10 נק']

נתון קוד חלקי של המחלקה Fraction. עליכם לכתוב קוד חדש, כך שכל עצם מהמחלקה:

```
Fraction<int>
```

יתנהג כמו Fraction הנתון, אבל יהיה ניתן גם לייצר עצמים מהטיפוס:

```
Fraction<short>
```

```
Fraction<long>
```

בלי לכתוב קוד נוסף. אין צורך לממש אופרטורים או פונקציות אחרות שלא מופיעות בקוד להלן. עליכם לכתוב קוד מלא גם אם חלקו העתקה מהקוד להלן.

```
class Fraction {
private:
    int nom, den;
public:
    Fraction(int nn, int dn=1): nom(nn), den(dn) { }

    int Nom() const {
        return nom;
    }

    int Den() const {
        return den;
    }

    Fraction operator+ (const Fraction& other) {
        int nn = nom * other.den + den * other.nom;
        int dd = den * other.den;
        return Fraction(nn, dd);
    }
};
```

בהצלחה!!!