



מס' תלמיד: _____

קמפוס: _____

למילוי ע"י הסטודנט

שנה"ל תשפ"ב, סמסטר ב', מועד א'
שאלון בחינה בקורס: סדנא ב ++C
מספר קורס: 150018

- **שם המרצה:** גרינברג נורית, ווגל ג'ויס, ויזן אריה, חגיז מורדי מרדכי, חדד אורטל, חייט אלכסנדר, טרנר אורי, יטיב אריה, ליטמן מאיר, מילסטון עדינה, נדלר חני, עמר אפרת, קדרון דוד, רוזנבליט אורית, רז שרה, שנדוביץ חיים
- **תאריך הבחינה:** ט"ז סיוון תשפ"ב 15 / 06 / 2022
- **משך הבחינה (בדקות):** 180
- **חומר עזר מותר לשימוש:** אין
- **מחשבון:** אסור

המבחן כולל שלושה חלקים עפ"י הפירוט הבא:

- חלק א':** 7 שאלות סגורות. יש לענות על 6 מתוכן. כל שאלה 7 נקודות, סה"כ 42 נקודות.
 - חלק ב':** 3 שאלות פתוחות קצרות. יש לענות על כולן. כל שאלה 10 נקודות, סה"כ 30 נקודות.
 - חלק ג':** 2 שאלות פתוחות. יש לענות על כולן. כל שאלה 16 נקודות, סה"כ 32 נקודות.
- הציון המקסימלי: 100**

- את התשובות יש לכתוב **בגוף המבחן** בלבד (תשובות במחברת או בדפי טיוטא לא ייבדקו).

תלמיד יקר,

1. **נוהל הבחינות של המרכז האקדמי לב מחייב אותך,** באחריותך לקוראו ולהכירו - בחינה עלולה להיפסל על כל חריגה מהנוהל.
2. אם אינך מבין את כוונת המרצה בשאלה כלשהי, עליך לכתוב בראש התשובה כיצד הינך מבין את השאלה ולפתור בהתאם. המרצה ישקול האם יש מקום להבנה זו ואז ינקד בהתאם.
3. **חובה להחזיר את השאלון.** מחברת שלא יצורף לה השאלון, לא תיבדק!
4. לידיעתך, תורדנה נקודות לא רק על שגיאות, אלא גם על תוספות לא רלוונטיות, העדר נימוק הולם לתשובה, חוסר סדר ותשובה דו-משמעית, כאשר נדרשת תשובה חד משמעית.

בהצלחה רבה !

לשימוש הבודקים בלבד:

חלק א	חלק ב			חלק ג		סה"כ
1 - 7	8	9	10	11	12	

שנה"ל תשפ"ב, סמסטר ב, מועד א

שאלון בחינה בקורס: סדנא ב ++C

מספר קורס: 150018

שאלון בקורס סדנא ב- ++C:

חלק א'

לפניך 7 שאלות קצרות. יש לענות על 6 מתוכן, 7 נקודות לכל שאלה (סה"כ 42 נקודות)
סמן בצורה ברורה איזו שאלה לא לבדוק ע"י מחיקה ב X של כל התשובות לשאלה זו.
במידה ותענה על כל השאלות, ייבדקו ויחושבו לציון הסופי 6 מתוכן באופן אקראי.

שאלה מספר 1:

לפניך תכנית המתייחסת למחלקה בשם Animal המייצגת בעל חיים, ומחלקה בשם Dog היורשת ממנה:

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

class Animal
{
public:
    Animal() { instances++; }

    Animal(int x) { instances += x; }

    static int instances;
};

class Dog : public Animal
{
public:
    Dog() {};
    Dog(bool isGuideDog) :m_isGuideDog(isGuideDog)
    {
        Animal a;
        m_myFriend = a;
    }
private:
    bool m_isGuideDog;
    Animal m_myFriend;
};

int Animal::instances = 0;

int main()
{
    Animal a1;
    Dog d1;
    Dog d2(true);
    cout << a1.instances;
}
```

מה יהיה פלט התכנית לאחר הרצת התכנית? (בעמוד הבא)

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב C++
מספר קורס: 150018

א. 6

ב. 4

ג. 5

ד. 3

ה. 7

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב C++
מספר קורס: 150018

שאלה מספר 2:

נתונה תכנית (בעמוד הבא. מספרי השורות נועדו לצורך השאלה, ואינם חלק מהתכנית)

התכנית כוללת הגדרות ושימוש בשלוש מחלקות: מחלקת בסיס Five, ושתי מחלקות היורשות ממנה.

סמן את התשובה הנכונה:

א. בשורה 12 במחלקה Four, ישנה שגיאת קומפילציה:

```
void print() override {  
    מכיוון שאסור היה להשתמש במילה override. יש לתקן כך:  
void print() {
```

ב. בשורה 22, במחלקה Three, ישנה שגיאת קומפילציה:

```
void print() {  
    מכיוון שהיה צריך להשתמש במילה override. יש לתקן כך:  
void print() override {
```

ג. בשורות 32-33 בתכנית הראשית, ישנה שגיאת זמן ריצה:

```
for (int i = 0; i < 2; ++i)  
    delete sentence[i];  
מכיוון שהיה צריך לבצע את שחרור הזיכרון בתוך ההורס (destructor), ולא בתכנית הראשית.  
יש לתקן על ידי כתיבת הורס עבור כל המחלקות.
```

ד. בשורה 31 בתכנית הראשית, ישנה שגיאת קומפילציה:

```
sentence[i]->print();  
מכיוון שישנו זימון לפונקציה print שאינה מוגדרת כך שהעצם עליו היא עובדת מוגן משינויים  
(const), בעוד ש-sentence מוגדר כקבוע. יש לתקן את שורות 6, 12, ו-22, ע"י הוספת const.  
כלומר, לשנות את שורה 6 כך:
```

```
virtual void print() const {  
    לשנות את שורה 12 כך:  
void print() override const {  
    ולשנות את שורה 22 כך:  
void print() const {
```

ה. התכנית תקינה ועוברת קומפילציה וריצה.

כמו כן, ניתן לשנות את שורה 6 כך:

```
virtual void print() = 0;  
ולא לשנות דבר בשאר המחלקות, והתכנית תישאר תקינה.
```

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב C++
מספר קורס: 150018

```
1. #include <iostream>
2. using namespace std;

3. class Five
4. {
5. public:
6.     virtual void print() {
7.         cout << 1;
8.     }
9. };

10. class Four : public Five {
11. public:
12.     void print() override {
13.         Five::print();
14.         cout << 2;
15.     }
16. };

17. class Three : public Five
18. {
19. private:
20.     Four a;

21. public:
22.     void print() {
23.         a.print();
24.         cout << 3;
25.     }
26. };

27. int main()
28. {
29.     const Five* sentence[2] = { new Four(), new Three() };

30.     for (int i = 0; i < 2; ++i)
31.         sentence[i]->print();

32.     for (int i = 0; i < 2; ++i)
33.         delete sentence[i];

34. }
```

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב C++
מספר קורס: 150018

שאלה מספר 3:

לפניך קטע קוד הכולל הגדרה של שתי מחלקות, A ו-B:

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

class A;

class B
{
public:
    void foo() ;
};

class A
{
    B b;
    int field;
};

void B:: foo()
{
    A a;
    cout << a.field;
}
```

קטע הקוד שלעיל, אינו עובר קומפילציה.

סמן את התשובה הנכונה ביותר:

א. כדי שקטע הקוד יעבור קומפילציה חובה להוסיף למחלקה A את השורה:
friend class B;
ואין צורך להוסיף כל דבר אחר.

ב. כדי שקטע הקוד יעבור קומפילציה חובה להוסיף למחלקה B את השורה:
friend class A;
ואין צורך להוסיף כל דבר אחר.

ג. כדי שקטע הקוד יעבור קומפילציה חובה להוסיף למחלקה B את השורה:
friend class A;
וגם חובה להוסיף למחלקה A את השורה:
friend class B;

ד. כדי שקטע הקוד יעבור קומפילציה חובה להוסיף למחלקה B את השורה:
friend int foo();

ה. כדי שקטע הקוד יעבור קומפילציה חובה להוסיף למחלקה A את השורה:
friend int foo();

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב C++
מספר קורס: 150018

שאלה מספר 4:

נתון קטע הקוד הבא, בו מוגדרת מחלקה A הכוללת מספר בנאים, 4 פונקציות גלובליות ותכנית ראשית:

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

class A {
public:
    A() { cout << "default" << endl; }
    A(const A & other) { cout << "copy" << endl; }
    A(const A && other) { cout << "move" << endl; }

};

void func1() {
    A a;
}
A func2() {
    A a;
    return a;
}
void func3(A a) {
}

void func4(A& a) {
}

int main() {
    A a1;
    A a2(a1);
    func1();
    func2();
    func3(a1);
    func4(a2);
}
```

מה יהיה פלט התכנית לאחר הרצת התכנית?

א.	ב.	ג.	ד.	ה.
default copy default default move copy	default move default default copy move	default copy default default	default copy default default copy copy copy	default copy default default copy copy

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב C++
מספר קורס: 150018

שאלה מספר 5:

נתון קטע הקוד הבא:

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

class A
{
public:
    void show1() { cout << "-A1-"; }
    virtual void show2() { cout << "-A2-"; }
};

class B : public A
{
public:
    void show1() { cout << "-B1-"; }
    void show2()
    {
        cout << "-B2-";
        A::show2();
    }
};

int main()
{
    try
    {
        B* pA = new B;
        throw pA;
    }

    catch (A* x)
    {
        x->show1();

        x->show2();
    }
}
```

סמן את התשובה הנכונה:

- א. התכנית תרוץ כראוי ותדפיס על המסך: -A1--B1--A2--B2-
- ב. התכנית תרוץ כראוי ותדפיס על המסך: -A1--A2-
- ג. התכנית תרוץ כראוי ותדפיס על המסך: -B1--B2--A2-
- ד. התכנית תרוץ כראוי ותדפיס על המסך: -A1--B2--A2-
- ה. התכנית תעוף במהלך הריצה בלי להדפיס דבר.

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב C++
מספר קורס: 150018

שאלה מספר 6:

נתונה התכנית הבאה, הבונה ומשתמשת בקובץ בשם fout.bin :

```
#include<iostream>
#include<fstream>
using namespace std;
struct Student
{
    char name[15];
    int mark;
};
int main()
{
    ofstream outfile("fout.bin");
    Student firstStudent;
    firstStudent.mark = 90;
    strcpy_s(firstStudent.name,4, "avi");

    outfile.write((char*)&firstStudent, sizeof(Student));
    outfile.close();

    ifstream infile("fout.bin");
    Student otherStudent;
    infile.read((char*)&otherStudent, sizeof(firstStudent));

    cout << otherStudent.name << endl;

    infile.close();

    return 0;
}
```

סמן את התשובה הנכונה :

- א. הקוד יעבור קומפילציה, אך יעוף בזמן ריצה בלי להדפיס דבר על המסך.
- ב. הקוד לא יעבור קומפילציה, כי לאחר שפתחנו קוד לצורך כתיבה, לא ניתן לפתוח את אותו הקובץ לקריאה.
- ג. הקוד תקין. הוא יעבור קומפילציה וירוצ, וידפיס על המסך avi.
- ד. הקוד לא יעבור קומפילציה, כי כשקראנו מהקובץ אסור היה לכתוב: sizeof(firstStudent). היה צריך לכתוב: sizeof(Student).
- ה. הקוד לא יעבור קומפילציה, כי כשכתבנו לקובץ אסור היה לכתוב: sizeof(Student). היה צריך לכתוב: sizeof(firstStudent).

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב ++C
מספר קורס: 150018

שאלה מספר 7:

נתונה הפונקציה f הבאה, המקבלת מספר ומחזירה ערך בוליאני:
(הבהרה: הערך $0x00000001$ הוא ביטוי בבסיס הקסדצימלי של הערך 1 העשרוני. המשמעות זהה – כלומר
היה ניתן לכתוב כאן: $x \& 1$, ולקבל תכנית זהה לחלוטין).

```
bool f(int x)
{
    unsigned int y = x & 0x00000001;
    y <<= 31;
    return y;
}
```

סמן את התשובה הנכונה:

- א. הפונקציה מחזירה true כאשר הפרמטר x הוא מספר שלילי, אחרת מחזירה false.
- ב. הפונקציה מחזירה true כאשר הפרמטר x הוא מספר אי זוגי, אחרת מחזירה false.
- ג. הפונקציה שגיאת קומפילציה (הידור).
- ד. הפונקציה מכניסה לתוך y את הערך 31, ולאחר מכן מחזירה true אם הצליח להכניס למשתנה, false אחרת.
- ה. הפונקציה תחזיר false עבור כל ערך שהיא מקבלת.

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב ++C
מספר קורס: 150018

חלק ב'

לפניך 3 שאלות פתוחות קצרות. יש לענות על כולן, 10 נקודות לכל שאלה (סה"כ 30 נקודות)

שאלה מספר 8:

בקטע הקוד הבא חסרות מספר שורות המסומנות בקו תחתי. עליך להשלימם לפי ההוראות בסוף השאלה.

לפניך קטע קוד הכולל הגדרה של רשומה (struct) לשמירת מידע על הטמפרטורה בחודש. בתכנית הראשית, מחזיקים מערך עבור 12 חודשי השנה, ומאתחלים אותו באמצעות הפעולה init, כך שהמידע עבור כל חודש כולל את שם החודש ואת מספר המעלות הממוצע באותו חודש. (הפעולה init לא מוצגת כאן, אך יש להניח שהיא פועלת כראוי). בשורות החסרות, יש לעדכן את התכונה isHot עבור כל חודש במערך months, כך שבמידה ומספר המעלות הממוצע בחודש (degree) היה מעל 30 מעלות, התכונה isHot תהיה true, אחרת false. במקביל (באותה פעולה) – יש לספור כמה חודשים במהלך השנה היו חמים (עם ממוצע מעל 30 מעלות) ולעדכן את המשתנה countHotMonths.

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

struct Month {
    string name; // שם החודש
    bool isHot; // האם מספר המעלות הממוצע בחודש היה מעל 30 מעלות
    double degrees; // מספר ממוצע של מעלות בחודש
};

void init(Month** months); //הפעולה תקינה, ולא מוצגת כאן

int main() {
    Month* months[12];

    init(months);

    int countHotMonths = 0;

    for_each(_____, _____,
             _____
             _____
             _____
             _____);

    cout << "the number of hot months is: " << countHotMonths;

}
```

שים לב – אין להתייחס לאורך הקו. השתמש במספר השורות שרצוי לך, אך אין להוסיף פקודות – אלא להשלים את השימוש בפקודת for_each בלבד.

שאלה מספר 9:

רוצים לבנות מחלקה Order לצורך טיפול בהזמנות מזון של לקוחות.

במחלקה יהיו שני שדות עבור כל לקוח: מספר זהות של הלקוח (id), וכן רשימת המאכלים (שם המאכל והקוד שלו בתפריט) שהלקוח הזמין (listOfFoods).

להלן ההצהרה של המחלקה Order:

```
class Order
{
public:
    class Food //מאכל
    {
    public:
        string name; //שם המאכל
        int code; // קוד המאכל
        Food(string myname, int mycode) :name(myname), code(mycode) {}
    };
    Order(int myid) :id(myid) {}
    void add(Food food);
    void print();

private:
    int id; // תז של הלקוח
    list <Food> listOfFoods; // רשימת המאכלים שהזמין
};
```

א. להלן המימוש עבור הפונקציה add שהוגדרה במחלקה. יש להשלים את הקוד עבור המתודה add בשורה המסומנת בקו, בכדי שהפונקציה תוסיף לרשימה את המאכל food.

```
void Order::add(Food food)
{
    _____
}
```

ב. להלן המימוש עבור הפונקציה print שהוגדרה במחלקה. יש להשלים את הקוד עבור המתודה print, בארבעת השורות המסומנות בקו, בכדי שהפונקציה תדפיס את רשימת כל המאכלים של הלקוח (קוד המאכל ושם המאכל, כל מאכל בשורה נפרדת). שים לב! אין להשתמש בפקודת auto לצורך הגדרת המשתנה it שבתוך הלולאה. יש להגדיר את הטיפוס במפורש.

```
void Order::print()
{
    for ( _____ it = _____ ;
         it != _____ ; it++)
        cout << _____ << endl;
}
```

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב C++
מספר קורס: 150018

שאלה מספר 10:

נתונה הגדרה חלקית של מחלקה בשם Arr (קיימות עוד מתודות ומימוש שלא מובאים כאן).
המחלקה פועלת כמערך של מספרים שלמים (מספרי השורות נועדו לצורך השאלה, ואינם חלק מהתכנית):

```
0.  
1. class Arr {  
2.     protected:  
3.         int* data;  
4.         int size;//size in use  
5.         int capacity;//available capacity  
6.     public:  
7.         Arr(int capacity = 0);  
8.         Arr(const Arr&);  
9.         virtual ~Arr();  
  
10.        Arr & operator = (const Arr &);  
11.        int& operator [](int index);  
12.        int getSize() const;  
13.        int getCapacity() const;  
14.        void insert(int value);  
15.        void clear();  
16.        int dellast();  
17. };
```

מעוניינים לשנות את המחלקה כך שתוכל לעבוד על כל טיפוס נתונים שהוא (מחלקה גנרית).
תקן את הנדרש בקטע הממשק שלעיל, כך שההגדרות תתאמנה למחלקה זוהה, אך גנרית.
(בשורות להלן יש להעתיק אך ורק את השורות שבהן יש שינוי – לפי מספר השורה המתאים, ולבצע את השינוי הנדרש)

```
0. _____  
1. _____  
2. _____  
3. _____  
4. _____  
5. _____  
6. _____  
7. _____  
8. _____  
9. _____  
10. _____  
11. _____  
12. _____  
13. _____  
14. _____  
15. _____  
16. _____  
17. _____
```

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב ++C
מספר קורס: 150018

חלק ג'

לפניך 2 שאלות פתוחות. יש לענות על כולן, 16 נקודות לכל שאלה (סה"כ 32 נקודות)

שאלה מספר 11:

בהרצאה ראינו את מחלקת List – מבנה נתונים של רשימה המכיל מספרים (ההצהרה והמימוש המלאים נמצאים בנספח א של מבחן זה).

א. הוסף למחלקה List מתודה בשם size.
על המתודה להחזיר את מספר האיברים השמורים ברשימה.

ב. הוסף למחלקה List מתודה בשם removeAll שמקבלת כפרמטר מצביע לפונקציה בוליאנית שבדקת את קיומו של תנאי כלשהו על איבר בודד מסוג מספר שלם (כגון: מתחלק ב-3, זוגי ועוד). המתודה removeAll מסירה מהרשימה את כל האיברים שעבורם התנאי מתקיים.

תשובה לשאלה 11:

כותרת ומימוש המתודה size :

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב C++
מספר קורס: 150018

כותרת ומימוש המתודה removeAll:

שאלה מספר 12:

יוסי, שהוא שף מצליח – פתח קייטרינג פרטי של "מאכלי השף". הוא מתמחה במאכלי בשר וחלב (בשני מטבחים נפרדים, כמובן), ומעוניין לתת ללקוחותיו שירות מיטבי. עבור כל לקוח שרוצה להזמין – יוסי יוצר הזמנה, בה הוא מוסיף לסל המוצרים את המאכלים שהלקוח מעוניין להזמין, ואת הכמות הרצויה מכל מאכל. לאחר מכן יוסי מחשב את העלות הכוללת של ההזמנה ומוציא חשבונית ללקוח.

לשם כך, עליך לסייע לו בכתיבת מחלקות מתאימות לפי ההנחיות להלן:

- i. מחלקת ChefsDishes ובה התכונות הבאות:
 - dish – קוד מספרי עבור המאכל
 - pricePerFood – שהוא המחיר לכל יחידה מהמאכל
 - amount – מספר היחידות שהלקוח רוצה להזמין מהמאכל.ערכי ברירת המחדל לכל התכונות הללו הוא 0. כמו כן – למחלקה זו יש לכתוב:
 - בנאי המקבל כפרמטרים את ערכי התכונות ויוצר עצם מאותחל בערכים הללו.
 - מתודה מופשטת (abstract, וירטואלית טהורה) פרטית בשם discountCalculation לחישוב מחיר סופי – כולל הנחה עבור המאכל (אם ישנה). המתודה מקבלת את המחיר שחושב עבור ההזמנה, ומחזירה את המחיר לאחר הנחה. במידה ואין הנחה – המתודה מחזירה את המחיר ללא הנחה.
 - מתודה בשם priceCalculation המחשבת ומחזירה את המחיר של כל פריט בהזמנה. המתודה מחשב את העלות כך: מחיר ליחידה כפול מספר היחידות, לאחר מכן המתודה מזמנת את המתודה discountCalculation עם העלות שחושבה, ומחזירה את המחיר הסופי.
- ii. מחלקת DairyFood עבור מאכלי חלב היורשת ממחלקת ChefsDishes. מחלקה זו, תכיל בנוסף לתכונות הקיימות (קוד מחיר וכמות):
 - תכונה בוליאנית בשם coupon המייצגת האם יש ללקוח קופון הנחה. ערך ברירת המחדל של הקופון הוא false.כמו כן, מחלקה זו תכיל:
 - בנאי המקבל ערכים
 - מימוש של המתודה discountCalculation. במידה ויש ללקוח קופון, הוא מקבל הנחה של 10% על העלות שחושבה. אחרת – משלם את העלות המלאה.
- iii. מחלקת MeatFood עבור מאכלי בשר היורשת ממחלקת ChefsDishes. מחלקה זו תכיל:
 - בנאי המקבל ערכים
 - מימוש של המתודה discountCalculation. במידה והלקוח הזמין מעל 5 יחידות, הוא יקבל הנחה של 15% על העלות שחושבה, אחרת – משלם את העלות המלאה.

לאחר מכן כתוב פונקציה גלובלית calculatePerOrder, המקבלת מערך בשם order של ChefsDishes המכיל את פרטי ההזמנה של לקוח, ואת גודלו (size) ומחזירה את המחיר הכולל שעל הלקוח לשלם עבור כל הפריטים.

הערות:

- אין צורך לבדוק תקינות קלט.
- אין צורך לכתוב בנאי ברירת מחדל.
- אין לתת ערכי ברירת מחדל לערכי הפרמטרים של הבנאי.
- הימנע משכפול קוד (למשל בבנאים). כלומר – אם יש כבר קוד המבצע את הפעולה, השתמש בו במקום לכתוב אותו שנית).
- את המתודות של המחלקות המכילות קוד, יש לממש מחוץ לממשק המחלקה.

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב C++
מספר קורס: 150018

תשובה לשאלה 12:

i. ממשק של מחלקת ChefsDishes:

מימוש של מחלקת ChefsDishes:

ii. ממשק של מחלקת DairyFood:

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב ++C
מספר קורס: 150018

מימוש של מחלקת DairyFood:

iii. ממשק של מחלקת MeatFood:

מימוש של מחלקת MeatFood:

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב ++C
מספר קורס: 150018

מימוש של פונקציה גלובלית calculatePerOrder:

מקום פנוי במידת הצורך להשלמת תשובות.
אנא ציין במפורש לאיזו שאלה התשובה שלהלן _____ ולאיזה סעיף. _____

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב C++
מספר קורס: 150018

נספח א' – ממשק ומימוש המחלקה List:

קובץ List.h

```
//-----  
// class List  
//      arbitrary size Lists  
//      permits insertion and removal  
//      only from the front of the List  
//-----  
class List  
{  
public:  
    //-----  
    // inner class link  
    // a single element for the linked List  
    //-----  
    class Link  
    {  
    public:  
        // constructor  
        Link(int linkValue, Link* nextPtr);  
        Link(const Link&);  
        // data areas  
        int value;  
        Link* next;  
    };    //end of class Link  
public:  
    // constructors  
    List();  
    List(const List&);  
    ~List();  
  
    // operations  
    void add(int value);  
    int firstElement() const;  
    bool search(const int& value) const;  
    bool isEmpty() const;  
    void removeFirst();  
    void clear();  
  
protected:  
    // data field  
    Link* head;  
};
```

שנה"ל תשפ"ב, סמסטר ב, מועד א
שאלון בחינה בקורס: סדנא ב C++
מספר קורס: 150018

קובץ List.cpp:

```
#include "List.h"

//-----
// class Link implementation
//-----
List::Link::Link(int val, Link* nxt) : value(val), next(nxt) {}

List::Link::Link(const Link& source) : value(source.value), next(source.next){}

//-----
// class List implementation
//-----
List::List() : head(nullptr)
{
    // no further initialization
}

List::List(const List& l)
{
    Link* src, * trg;
    if (l.head == nullptr)
        head = nullptr;
    else
    {
        head = new Link((l.head)->value, nullptr);
        src = l.head;
        trg = head;
        while (src->next != nullptr)
        {
            trg->next = new Link((src->next)->value, nullptr);
            src = src->next;
            trg = trg->next;
        }
    }
}

List::~List()
{
    //if(!head)
    clear();
}

void List::clear() {
    // empty all elements from the List
    Link* next;
    for (Link* p = head; p != nullptr; p = next)
    {
        // delete the element pointed to by p
        next = p->next;
        p->next = nullptr;
        delete p;
    }
    // mark that the List contains no elements
    head = nullptr;
}

bool List::isEmpty() const
{
    // test to see if the List is empty
}
```

```
// List is empty if the pointer to the head
// Link is null

return head == nullptr;
}

void List::add(int val)
{
    //Add a new value to the front of a Linked List
    if (isEmpty())
        head = new Link(val, nullptr);
    else {
        Link* temp = new Link(val, head);
        head = temp;
    }
}

int List::firstElement() const
{
    // return first value in List
    if (isEmpty())
        throw "the List is empty, no first Element";
    return head->value;
}

bool List::search(const int& val) const
{
    // loop to test each element
    for (Link* p = head; p != nullptr; p = p->next)
        if (val == p->value)
            return true;

    // not found
    return false;
}

void List::removeFirst()
{
    // make sure there is a first element
    if (isEmpty())
        throw "the List is empty, no Elements to remove";
    // save pointer to the removed node
    Link* p = head;
    // reassign the first node
    head = p->next;
    if (head != nullptr)
        p->next = nullptr;
    // recover memory used by the first element
    delete p;
}
```