



פקולטה: מדעי הטבע
מחלקה: מדעי המחשב
שם הקורס: תכנות מתקדם ושפת C++
קוד הקורס: 7038510
תאריך הבחינה: 15/09/2019
סמסטר: ב, מועד: 3
משך הבחינה: שעתיים
שם המרצה: ד"ר פנחס ויסברג

יש לענות על כל השאלות
ללא חומר עזר

לפניכם 10 שאלות, משקל כל שאלה 10 נקודות

בהצלחה

[Info]

make

המחלקה Date מוגדרת בקובץ date.h וממומשת בקובץ date.cpp
הקובץ main.cpp משתמש במחלקה Date
איזה קובץ makefile יקמפל את הקבצים וייצור קובץ ריצה בשם date ?

[q1]

הקובץ :

date: main.cpp date.cpp date.h
g++ -o date main.cpp date.cpp

[a]

הקובץ :

a.out: main.cpp date.cpp date.h
g++ -o date main.cpp date.cpp date.h

[a]

הקובץ :

date: main.cpp date.cpp date.h
g++ -o date main.cpp date.cpp date.h

[a]

הקובץ :

date: main.cpp date.cpp
g++ -o date main.cpp date.cpp

[a]

Creating a Class

מהו הקוד הנכון להשמת ההעתקה (Copy assignment) עבור המחלקה Vector שהגדרנו :

```
class Vector {  
    int sz;  
    double* elem;
```

[q2]

הקוד הנכון הוא :

```
Vector& Vector::operator=(const Vector& rhs)  
{  
    double* p = new double[rhs.sz];  
    copy(rhs.elem, rhs.elem+rhs.sz, p);  
    delete[] elem;  
    elem = p;  
    sz = rhs.sz;  
    return *this;  
}
```

[a]

הקוד הנכון הוא :

```
Vector& Vector::operator=(const Vector& rhs)  
{  
    elem = new double[rhs.sz];  
    copy(rhs.elem, rhs.elem+rhs.sz, elem);  
    sz = rhs.sz;  
    return *this;  
}
```

[a]

הקוד הנכון הוא :

```
Vector& Vector::operator=(const Vector& rhs)  
{  
    double* p = new double[rhs.sz];  
    copy(rhs.elem, rhs.elem+rhs.sz, p);  
    elem = p;  
    delete[] p;  
    sz = rhs.sz;  
    return *this;  
}
```

[a]

הקוד הנכון הוא :

```
Vector& Vector::operator=(const Vector& rhs)  
{  
    double* p = new double[sz];  
    copy(rhs.elem, rhs.elem+rhs.sz, p);  
    delete[] elem;  
    elem = p;  
    return *this;  
}
```

[a]

Creating a Class

מהו הקוד הנכון להשמת הזזה (Move assignment) עבור המחלקה Vector שהגדרנו :

```
class Vector {  
    int sz;  
    double* elem;
```

[q3]

הקוד הנכון הוא :

```
Vector& Vector::operator=(Vector&& a)  
{  
    delete[] elem;  
    elem = a.elem;  
    sz = a.sz;  
    a.elem = nullptr;  
    a.sz = 0;  
    return *this;  
}
```

[a]

הקוד הנכון הוא :

```
Vector &Vector::Vector(Vector&& a)  
    :sz{a.sz}, elem{a.elem}  
{  
    a.sz = 0;  
    a.elem = nullptr;  
    return *this;  
}
```

[a]

הקוד הנכון הוא :

```
Vector& Vector::operator=(Vector&& a)  
{  
    elem = a.elem;  
    sz = a.sz;  
    a.elem = nullptr;  
    a.sz = 0;  
    return *this;  
}
```

[a]

הקוד הנכון הוא :

```
Vector& Vector::operator=(Vector&& a)  
{  
    delete[] elem;  
    elem = a.elem;  
    sz = a.sz;  
    a.sz = 0;  
    return *this;  
}
```

[a]

STL

מה יהיה תוכן הווקטור vec לאחר ביצוע הפקודות הבאות:

```
vector<int> vec{5, 5, 22, 22, 8, 8, 8, 4};
vector<int> vec2;
unique_copy(begin(vec) + 1, end(vec) - 1, back_inserter(vec2));
copy(begin(vec2), end(vec2), begin(vec));
```

[q4]

{5, 22, 8, 22, 8, 8, 8, 4}

[a]

{5, 5, 22, 22, 8, 8, 8, 4, 5, 22, 8}

[a]

{5, 22, 8, 4, 8, 8, 8, 4}

[a]

{6 22 8 4 8 8 8 3}

[a]

Lambda Function

איזה משורות הקוד הבאות תכתוב את סכום אברי הווקטור לתוך sum ?

[q5]

שורות הקוד :

```
int sum = 0;
for_each(begin(vec), end(vec), [&sum] (int x) { sum += x; });
```

[a]
שורות הקוד :

```
int sum = 0;
for_each(begin(vec), end(vec), [sum] (int &x) { sum += x; });
```

[a]
שורות הקוד :

```
int sum = 0;
for_each(begin(vec), end(vec), [x&] (int sum) { sum += x; });
```

[a]
שורות הקוד :

```
int sum = 0;
for_each(begin(vec), end(vec), [x] (int& sum) { sum += x; });
```

[a]

Query Program

מהם שתי השורות החסרות בקוד ?

```
class TextQuery {
public:
    using line_no = std::vector<std::string>::size_type;
    TextQuery(std::ifstream&);
    QueryResult query(const std::string&) const;
private:
    std::shared_ptr<std::vector<std::string>> file;
    std::map<std::string, std::shared_ptr<std::set<line_no>>> wm;
};

TextQuery::TextQuery(std::ifstream &is): file(new vector<string>)
{
    string text;
    while (getline(is, text)) {
        file->push_back(text);
        int n = file->size() - 1;
        istringstream line(text);
        string word;
        while (line >> word) {
            -----
            -----
            lines.reset(new set<line_no>);
            lines->insert(n);
        }
    }
}
```

[q6]

השורות החסרות הן :

```
auto &lines = wm[word];
if (!lines)
```

[a]

השורות החסרות הן :

```
auto &lines = wm[n];
if (!lines)
```

[a]

השורות החסרות הן :

```
auto lines = wm[word];
if (!lines)
```

[a]

השורות החסרות הן :

```
auto &lines = wm[word];
if (lines)
```

[a]

ביטויים רגולריים

מה הפלט של התוכנית הבאה:

```
string str = "ABCABCXYXYXY";
regex reg("([XY]+)(XYZ)*");
smatch match;
if (regex_search(str, match, reg))
    cout << match[1] << '\n';
else cout << "not found\n";
```

[q7]

XYXYXY

[a]

XY

[a]

X

[a]

not found

[a]

זיכרון וירטואלי

כמה כניסות יהיו בטבלת הדפים אם מרחב הזיכרון הווירטואלי הוא 32 ביטים וגודל כל דף הוא
!16KB

[q8]

262,144

[a]

8,192

[a]

131,072

[a]

16,384

[a]

זיכרון דינמי

בניהול זיכרון דינמי בשיטת Implicit Free List:

- א. זמן ההקצאה של בלוק תלוי במספר הבלוקים
- ב. זמן ההקצאה של בלוק תלוי במספר הבלוקים הפנויים
- ג. זמן מיזוג בלוק עם הבלוק שלפניו תלוי במספר הבלוקים
- ד. זמן מיזוג בלוק עם הבלוק שלפניו הוא קבוע

[q9]

משפט א ומשפט ד נכונים

[a]

משפט א ומשפט ג נכונים

[a]

משפט ב ומשפט ד נכונים

[a]

משפט א ומשפט ג נכונים

[a]

זיכרון דינמי

בניהול זיכרון דינמי בשיטת Implicit Free List:

- א. בשחרור רשימה מעגלית, לא תתכן דליפת זיכרון כשמשתמשים ב- shared pointer
- ב. המשתנים המוגדרים בתוך פונקציה יכולים להכיל root set של garbage collection
- ג. המשתנים המוגדרים מחוץ לפונקציה יכולים להכיל root set של garbage collection
- ד. shared_ptr יותר מהיר מ- unique_ptr

[q10]

משפט ב ומשפט ג נכונים

[a]

משפט א ומשפט ד נכונים

[a]

משפט א ומשפט ב נכונים

[a]

משפט ג ומשפט ד נכונים

[a]