

TJT-L33 Olio-ohjelmointi, JY/TKTL, syksy 2002

5. demonstraatiotehtävät, viikko 44

2-4: Suunnittele ja toteuta kuvatuunlainen luokkajärjestelmä. Kielenä voit käyttää C++:aa tai Javaa, tai pakon edessä jotain muuta oliokieltä. Muista myös kuvata ratkaisusi UML:llä. Käsiniirretykin kelpaa. :)

1. Alla on annettu pääohjelma, joka käyttää luokkia Yliluokka ja Aliluokka.

```
#include <iostream>
using std::cout;
using std::endl;
using std::ostream;

#include <string>
using std::string;

#include "Aliluokka.h"
#include "Yliluokka.h"

int main(void) {
    Yliluokka yli;
    Aliluokka ali;
    Aliluokka *alios=&yli;

    cout << " ali.tee " << ali.tee()
         << " ali.virtu " << ali.virtu() << endl;
    cout << " yli.tee " << yli.tee()
         << " yli.virtu " << yli.virtu() << endl;
    cout << " alios.tee " << alios->tee()
         << " alios.virtu " << alios->virtu() << endl;

    return 0;
}
```

Toteuta luokat Yliluokka ja Aliluokka niin, että ohjelma tulostaa seuraavan (osoitteet tietty vaihtuvat):

```
Syntyi Aliluokka osoitteeseen 0xbffffc50.
Syntyi Yliluokka osoitteeseen 0xbffffc50.
Syntyi Aliluokka osoitteeseen 0xbffffc40.
ali.tee Aliluokka::tee ali.virtu Aliluokka::virtu
yli.tee Yliluokka::tee yli.virtu Yliluokka::virtu
alios.tee Aliluokka::tee alios.virtu Yliluokka::virtu
Kuoli Aliluokka osoitteesta 0xbffffc40.
```

Kuoli Yliluokka osoitteesta 0xbffffc50.
Kuoli Aliluokka osoitteesta 0xbffffc50.

2. Toteuta luokka, joka varastoi merkkijonoja (C++-standardin luokka `string`). Luokassa tulee olla metodit alkion lisäämiseen varaston "loppuun", halutun alkion arvon hakemiseen (indeksointi), halutun alkion poistamiseen, varaston koon selvittämiseen. Varaston tulee kasvaa tarvittaessa, mutta sen ei tarvitse pienetä. C++-standardin säilöluokkia ei saa käyttää.
3. Muuta yllätoteuttamaasi luokkaa siten, että sillä voi varastoida mitä tahansa abstraktista luokasta *Varasto_olio* perittyjen luokkien olioita. *Varasto_olio*-luokan määrittely olkoon:

```
class Varasto_olio {  
public:  
    virtual string tyyppi() = 0;  
    // palauttaa periytetyn luokan nimen  
};
```

4. Jatkaen viime viikon PehmoÖllerö-linjalla:
Suunnittele ja toteuta luokkarakenne, jolla voit säilöä tietoa n-puussa, eli puun alkiolla voi olla vaihteleva määrä alialkioita. Alkio on siten joko lehti tai solmu. Tyypiesimerkki tällaisesta rakenteesta on käyttöjärjestelmän tiedostorakenne: Hakemisto voi sisältää tiedostojen lisäksi myös toisia hakemistoja, jotka voivat sisältää tiedostojen lisäksi myös toisia hakemistoja, jotka ...
Kun puun jollekin alkiolle kutsutaan metodia `tee()`, tulee sen joko alkion ollessa puun lehti toteuttaa metodi, tai alkion ollessa solmu toteuttaa `tee()`-metodi jokaiselle alialkiolle.