Ohjelmistotekniikka

Matti Luukkainen, Jaako Iisala ja iso lauma muita ohjaajia

13.3.2023

Luokkakaaviot

- Todo-sovelluksen oleellista tietosisältöä kuvaavat luokat

```
class Todo:
    def __init__(self, content, done=False, user):
        self.content = content
        self.done = done
        self.user = user
        self.id = str(uuid.uuid4())

class User:
    def __init__(self, username, password):
        self.username = username
        self.password = password
```

Sekvenssikaavio

Mitä tapahtuu, kun maksukortilla jolla on rahaa 3 euroa, ostataan edullinen lounas?

Sekvenssikaavio

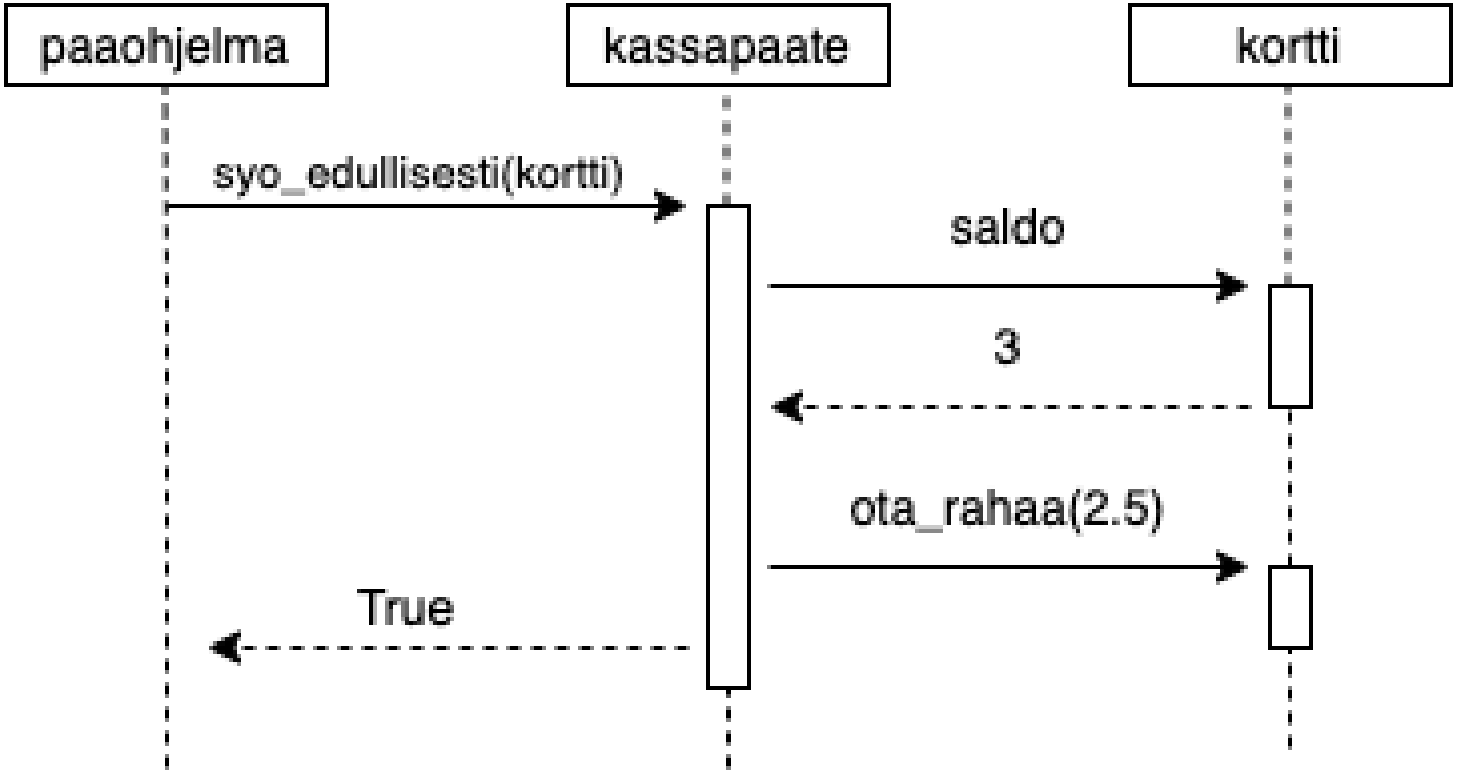
Mitä tapahtuu, kun maksukortilla jolla on rahaa 3 euroa, ostataan edullinen lounas?

```
class Kassapaate:
    def __init__(self):
        self.EDULLISEN_HINTA = 2.5

    def syo_edullisesti(self, kortti: Maksukortti):
        if kortti.saldo < self.EDULLISEN_HINTA:
            return False

        kortti.ota_rahaa(self.EDULLISEN_HINTA):
        self.edulliset += 1
        return True
```

Onnistunut ostos sekvenssikaaviona



- Oliot ovat laatikoita joista lähtee alas “elämänlanka”
- Aika etenee ylhäältä alas
- Metodikutsut ovat nuolia, jotka yhdistävää kutsuvan ja kutsutun olion elämänlangat
- Paluuarvo merkitään katkoviivalla

HARJOITUSTYÖ

Yleistä

- Kurssin pääpainon muodostaa viikolla 2 aloitettava harjoitustyö
- Harjoitustyössä toteutetaan itsenäisesti ohjelmisto omavalintaisesta aiheesta

Yleistä

- Kurssin pääpainon muodostaa viikolla 2 aloitettava harjoitustyö
- Harjoitustyössä toteutetaan itsenäisesti ohjelmisto omavalintaisesta aiheesta

- Harjoitustyötä tehdään itsenäisesti, mutta tarjolla on pajaohjausta
- Pajaa kampuksella (BK107)
  - ks kurssisivu
- Myös kurssin *Discordissa* voi kysellä apua ongelmatilanteissa

Älä plagioi

- Kurssilla seurataan Helsingin yliopiston opintokäytäntöjä
- Plagiarismi ja opintovilppi, eli esimerkiksi netissä olevien tai kaverilta saatujen vastausten kopiointi ja niiden palauttaminen omana työnä on kiellettyä
- Todettu opintovilppi johtaa kurssisuorituksen hylkäämiseen ja toistuva opintovilppi voi johtaa opinto-oikeuden määräaikaiseen menettämiseen

Älä plagioi

- Kurssilla seurataan Helsingin yliopiston opintokäytäntöjä
- Plagiarismi ja opintovilppi, eli esimerkiksi netissä olevien tai kaverilta saatujen vastausten kopiointi ja niiden palauttaminen omana työnä on kiellettyä
- Todettu opintovilppi johtaa kurssisuorituksen hylkäämiseen ja toistuva opintovilppi voi johtaa opinto-oikeuden määräaikaiseen menettämiseen

- ChatGPT:n käyttö on ok, mutta kaikki mihin ChatGPT:tä on käytetty (esim. jos sen generoimaa koodia on jossain), se tulee kertoa dokumentaatiossa, muuten kyseessä on opintovilppi.

Työn eteneminen

- Edetään viikottaisten tavoitteiden mukaan
- Työ on saatava valmiiksi kurssin aikana ja sitä on toteutettava tasaisesti, muuten kurssi katsotaan keskeytetyksi

Työn eteneminen

- Edetään viikottaisten tavoitteiden mukaan
- Työ on saatava valmiiksi kurssin aikana ja sitä on toteutettava tasaisesti, muuten kurssi katsotaan keskeytetyksi
- Samaa ohjelmaa ei voi jatkaa seuraavalla kurssilla (eli syksyllä 2023), vaan työ on aloitettava uudella aiheella alusta

Työn eteneminen

- Edetään viikottaisten tavoitteiden mukaan
- Työ on saatava valmiiksi kurssin aikana ja sitä on toteutettava tasaisesti, muuten kurssi katsotaan keskeytetyksi
- Samaa ohjelmaa ei voi jatkaa seuraavalla kurssilla (eli syksyllä 2023), vaan työ on aloitettava uudella aiheella alusta
- Koko kurssin arvostelu perustuu pääasiassa harjoitustyöstä saataviin pisteisiin
- Osa pisteistä kertyy viikoittaisten välitavoitteiden kautta, osa taas perustuu työn lopulliseen palautukseen

Viikkopalautukset ja palaute

- Kurssilla siis viikoittaiset palautukset, deadline tiistaisin klo 23.59:59
  - näistä kertyy osa kurssin pistestä (17/60)
- Viikkopalautuksista saa pistemäärän lisäksi pienen palautteen

Viikkopalautukset ja palaute

- Kurssilla siis viikoittaiset palautukset, deadline tiistaisin klo 23.59:59
  - näistä kertyy osa kurssin pistestä (17/60)
- Viikkopalautuksista saa pistemäärän lisäksi pienen palautteen
- Jos haluat harjoitustyöstä “laajempaa” palautetta
  - onko koodin rakenne hyvä
  - onko työ tarpeeksi laaja jotta siitä voisi saada arvosanan x
- **Viikkopalautteet eivät ole tähän tarkoitettu**

Viikkopalautukset ja palaute

- Kurssilla siis viikoittaiset palautukset, deadline tiistaisin klo 23.59:59
  - näistä kertyy osa kurssin pistestä (17/60)
- Viikkopalautuksista saa pistemäärän lisäksi pienen palautteen
- Jos haluat harjoitustyöstä “laajempaa” palautetta
  - onko koodin rakenne hyvä
  - onko työ tarpeeksi laaja jotta siitä voisi saada arvosanan x
- **Viikkopalautteet eivät ole tähän tarkoitettu**
- Mene pajaan
- Varaa aika “ohjaajan koodikatselmointiin”
  - 30 minuutin aika missä EI ole tarkoitus ratkaista koodaukseen liittyviä ongelmia vaan saada koodista laadullista palautetta

Viikkopalautukset ja palaute

- Kurssilla siis viikoittaiset palautukset, deadline tiistaisin klo 23.59:59
  - näistä kertyy osa kurssin pistestä (17/60)
- Viikkopalautuksista saa pistemäärän lisäksi pienen palautteen
- Jos haluat harjoitustyöstä “laajempaa” palautetta
  - onko koodin rakenne hyvä
  - onko työ tarpeeksi laaja jotta siitä voisi saada arvosanan x
- **Viikkopalautteet eivät ole tähän tarkoitettu**
- Mene pajaan
- Varaa aika “ohjaajan koodikatselmointiin”
  - 30 minuutin aika missä EI ole tarkoitus ratkaista koodaukseen liittyviä ongelmia vaan saada koodista laadullista palautetta
- Ohjaajan katselmoinnit alkavat noin kurssin neljännellä viikolla

Kieli

- Harjoitustyön ohjelmointikieli on Python
  - myös Java mahdollinen, kurssin Java-materiaalia ei kuitenkaan enää ylläpidetä, eli Javaa ei suositella...
- Ohjelmakoodin muuttujat, luokat ja metodit **kirjoitetaan englanniksi**
- Dokumentaatio voidaan kirjoittaa joko suomeksi tai englanniksi

Kieli

- Harjoitustyön ohjelmointikieli on Python
  - myös Java mahdollinen, kurssin Java-materiaalia ei kuitenkaan enää ylläpidetä, eli Javaa ei suositella...
- Ohjelmakoodin muuttujat, luokat ja metodit **kirjoitetaan englanniksi**
- Dokumentaatio voidaan kirjoittaa joko suomeksi tai englanniksi
- Web-sovelluksia kurssilla ei sallita
  - Sovelluksessa voi toki olla webissä toimivia komponentteja, mutta sovelluksen käyttöliittymän tulee olla ns. desktop-sovellus

Ohjelman toteutus

- Toteutus etenee “iteratiivisesti ja inkrementaalisesti”
  - Heti ensimmäisellä viikolla toteutetaan pieni käyttökelpoinen osa toiminnallisuudesta
  - ohjelman ydin pidetään koko ajan toimivana, uutta toiminnallisuutta lisäten, kunnes tavoiteltu laajuus on saavutettu

Ohjelman toteutus

- Toteutus etenee “iteratiivisesti ja inkrementaalisesti”
  - Heti ensimmäisellä viikolla toteutetaan pieni käyttökelpoinen osa toiminnallisuudesta
  - ohjelman ydin pidetään koko ajan toimivana, uutta toiminnallisuutta lisäten, kunnes tavoiteltu laajuus on saavutettu
- Iteratiiviseen tapaan tehdä ohjelma liittyy kiinteästi automatisoitu testaus
- Uutta toiminnallisuutta lisättäessä ja vanhaa muokatessa täytyy varmistua, että kaikki vanhat ominaisuudet toimivat edelleen

Ohjelman toteutus

- Toteutus etenee “iteratiivisesti ja inkrementaalisesti”
  - Heti ensimmäisellä viikolla toteutetaan pieni käyttökelpoinen osa toiminnallisuudesta
  - ohjelman ydin pidetään koko ajan toimivana, uutta toiminnallisuutta lisäten, kunnes tavoiteltu laajuus on saavutettu
- Iteratiiviseen tapaan tehdä ohjelma liittyy kiinteästi automatisoitu testaus
- Uutta toiminnallisuutta lisättäessä ja vanhaa muokatessa täytyy varmistua, että kaikki vanhat ominaisuudet toimivat edelleen
- *Jotta ohjelmaa pystyisi testaamaan, on tärkeää että sovelluslogiikkaa ei kirjoiteta käyttöliittymän sekaan*

Ohjelman toteutus

- Toteutus etenee “iteratiivisesti ja inkrementaalisesti”
  - Heti ensimmäisellä viikolla toteutetaan pieni käyttökelpoinen osa toiminnallisuudesta
  - ohjelman ydin pidetään koko ajan toimivana, uutta toiminnallisuutta lisäten, kunnes tavoiteltu laajuus on saavutettu
- Iteratiiviseen tapaan tehdä ohjelma liittyy kiinteästi automatisoitu testaus
- Uutta toiminnallisuutta lisättäessä ja vanhaa muokatessa täytyy varmistua, että kaikki vanhat ominaisuudet toimivat edelleen
- *Jotta ohjelmaa pystyisi testaamaan, on tärkeää että sovelluslogiikkaa ei kirjoiteta käyttöliittymän sekaan*
- Graafiseen käyttöliittymään suositellaan Pythonilla Tkinteriä tai Pygamea ja Javalla JavaFX:ää
- Tiedon talletus joko tiedostoon tai tietokantaan suositeltavaa

Ohjelman toteutus

- Tavoitteena on tuottaa ohjelma, joka voitaisiin antaa toiselle opiskelijalle ylläpidettäväksi ja täydennettäväksi
  - koodin on siis oltava ymmärrettävää ja jatkokehitityksen mahdollistavaa

Ohjelman toteutus

- Tavoitteena on tuottaa ohjelma, joka voitaisiin antaa toiselle opiskelijalle ylläpidettäväksi ja täydennettäväksi
  - koodin on siis oltava ymmärrettävää ja jatkokehitityksen mahdollistavaa
- Lopullisessa palautuksessa on oltava lähdekoodin lisäksi dokumentaatio ja automaattiset testit sekä Java-sovelluksissa jar-tiedosto

Ohjelman toteutus

- Tavoitteena on tuottaa ohjelma, joka voitaisiin antaa toiselle opiskelijalle ylläpidettäväksi ja täydennettäväksi
  - koodin on siis oltava ymmärrettävää ja jatkokehitityksen mahdollistavaa
- Lopullisessa palautuksessa on oltava lähdekoodin lisäksi dokumentaatio ja automaattiset testit sekä Java-sovelluksissa jar-tiedosto
- Toivottava dokumentaation taso käy ilmi referenssisovelluksesta https://github.com/ohjelmistotekniikka-hy/python-todo-app

Ohjelman toteutus

- Tavoitteena on tuottaa ohjelma, joka voitaisiin antaa toiselle opiskelijalle ylläpidettäväksi ja täydennettäväksi
  - koodin on siis oltava ymmärrettävää ja jatkokehitityksen mahdollistavaa
- Lopullisessa palautuksessa on oltava lähdekoodin lisäksi dokumentaatio ja automaattiset testit sekä Java-sovelluksissa jar-tiedosto
- Toivottava dokumentaation taso käy ilmi referenssisovelluksesta https://github.com/ohjelmistotekniikka-hy/python-todo-app
- Voit ottaa mallia referenssisovelluksen dokumentaatiosta mutta **älä copypastea**, se johtaa hylkäämiseen

Koodin laatuvaatimukset

- Kurssin tavoitteena on, että tuotoksesi voisi ottaa kuka tahansa kaverisi tai muu opiskelija ylläpidettäväksi ja laajennettavaksi
- Lopullisessa palautuksessa tavoitteena on *Clean code* eli selkeä, ylläpidettävä ja toimivaksi automatisoidusti testattu koodi

Koodin laatuvaatimukset

- Kurssin tavoitteena on, että tuotoksesi voisi ottaa kuka tahansa kaverisi tai muu opiskelija ylläpidettäväksi ja laajennettavaksi
- Lopullisessa palautuksessa tavoitteena on *Clean code* eli selkeä, ylläpidettävä ja toimivaksi automatisoidusti testattu koodi
- Jos käytät Pygamea, tyyli *ei voi olla sama* kuin Ohjelmoinnin jatkokurssin pygamemateriaalissa
  - ks https://ohjelmistotekniikka-hy.github.io/python/pygame

Hyvän aiheen ominaisuudet

- **Itseäsi kiinnostava aihe**
- Riittävän mutta ei liian laaja
  - Vältä eeppisiä aiheita, aloita riittävän pienestä
  - Valitse aihe, jonka perustoiminnallisuuden saa toteutettua nopeasti, mutta jota saa myös laajennettua helposti
  - Hyvässä aiheessa on muutamia logiikkaluokkia, tiedostojen tai tietokannan käsittelyä ja sovelluslogiikasta eriytetty käyttöliittymä

Hyvän aiheen ominaisuudet

- **Itseäsi kiinnostava aihe**
- Riittävän mutta ei liian laaja
  - Vältä eeppisiä aiheita, aloita riittävän pienestä
  - Valitse aihe, jonka perustoiminnallisuuden saa toteutettua nopeasti, mutta jota saa myös laajennettua helposti
  - Hyvässä aiheessa on muutamia logiikkaluokkia, tiedostojen tai tietokannan käsittelyä ja sovelluslogiikasta eriytetty käyttöliittymä
- Kurssilla pääpaino on
  - Toimivuus ja varautuminen virhetilanteisiin
  - Luokkien vastuut
  - Ohjelman selkeä rakenne
  - Laajennettavuus ja ylläpidettävyys

Hyvän aiheen ominaisuudet

- **Itseäsi kiinnostava aihe**
- Riittävän mutta ei liian laaja
  - Vältä eeppisiä aiheita, aloita riittävän pienestä
  - Valitse aihe, jonka perustoiminnallisuuden saa toteutettua nopeasti, mutta jota saa myös laajennettua helposti
  - Hyvässä aiheessa on muutamia logiikkaluokkia, tiedostojen tai tietokannan käsittelyä ja sovelluslogiikasta eriytetty käyttöliittymä
- Kurssilla pääpaino on
  - Toimivuus ja varautuminen virhetilanteisiin
  - Luokkien vastuut
  - Ohjelman selkeä rakenne
  - Laajennettavuus ja ylläpidettävyys
- **Tällä kurssilla ei ole tärkeää:**
  - Tekoäly
  - Grafiikka
  - Tietoturva
  - Tehokkuus

Huonon aiheen ominaisuuksia

- Kannattaa yrittää välttää aiheita, joissa pääpaino on tiedon säilömisessä tai liian monimutkaisessa käyttöliittymässä
- Paljon tietoa säilövät, esim. yli 3 tietokantataulua tarvitsevat sovellukset sopivat yleensä paremmin kurssille Tietokantasovellus
- Käyttöliittymäkeskeisissä aiheissa voi olla vaikea keksiä sovelluslogiikkaa, joka on enemmän tämän kurssin painopiste

Esimerkkejä aiheista

- Hyötyohjelmat
  - Aritmetiikan harjoittelua
  - Tehtävägeneraattori, joka antaa käyttäjälle tehtävän sekä mallivastauksen (esim. matematiikkaa, fysiikkaa, kemiaa, ...)
  - Code Snippet Manageri
  - Laskin, funktiolaskin, graafinen laskin
  - Budjetointisovellus
  - Opintojen seurantasovellus
  - HTML WYSIWYG-editor (What you see is what you get)

Esimerkkejä aiheista

- Reaaliaikaiset pelit
  - Tetris
  - Pong
  - Pacman
  - Tower Defence
  - Asteroids
  - Space Invaders
  - Yksinkertainen tasohyppypeli, esimerkiksi The Impossible Game

Esimerkkejä aiheista

- Vuoropohjaiset pelit
  - Tammi
  - Yatzy
  - Miinaharava
  - Laivanupotus
  - Yksinkertainen roolipeli tai luolastoseikkailu
  - Sudoku
  - Muistipeli
  - Ristinolla (mielivaltaisen kokoisella ruudukolla?)

Esimerkkejä aiheista

- Korttipelit
  - En Garde
  - Pasianssi
  - UNO
  - Texas Hold'em
- Omaan tieteenalaan, sivuaineeseen tai harrastukseen liittyvät hyötyohjelmat
  - Yksinkertainen fysiikkasimulaattori
  - DNA-ketjujen tutkija
  - Keräilykorttien hallintajärjestelmä
  - Fraktaaligeneraattori

Arvosteluperusteet tarkemmin

- Kurssin maksimi on 60 pistettä
- Ennen loppupalautusta jaossa 19 pistettä
  - Viikkodeadlinet 17p
  - Koodikatselmointi 2p

Arvosteluperusteet tarkemmin

- Kurssin maksimi on 60 pistettä
- Ennen loppupalautusta jaossa 19 pistettä
  - Viikkodeadlinet 17p
  - Koodikatselmointi 2p
- Loppupalautus ratkaise 41 pisteen kohtalon
  - Dokumentaatio 12p
  - Automatisoitu testaus 5p
  - Lopullinen ohjelma 24p
    - Laajuus, ominaisuudet ja koodin laatu

Arvosteluperusteet tarkemmin

- Kurssin maksimi on 60 pistettä
- Ennen loppupalautusta jaossa 19 pistettä
  - Viikkodeadlinet 17p
  - Koodikatselmointi 2p
- Loppupalautus ratkaise 41 pisteen kohtalon
  - Dokumentaatio 12p
  - Automatisoitu testaus 5p
  - Lopullinen ohjelma 24p
    - Laajuus, ominaisuudet ja koodin laatu
- Arvosanaan 1 riittää 30 pistettä, arvosanaan 5 tarvitaan noin 55 pistettä
- Läpipääsyyn vaatimuksena on lisäksi vähintään 10 pistettä lopullisesta ohjelmasta

Harjoitustyön vaikutus kurssipisteisiin

- käyttöliittymä 4p
  - 0p yksinkertainen tekstikäyttöliittymä
  - 1-2p monimutkainen tekstikäyttöliittymä
  - 2-3p yksinkertainen graafinen käyttöliittymä
  - 4p laaja graafinen käyttöliittymä

Harjoitustyön vaikutus kurssipisteisiin

- käyttöliittymä 4p
  - 0p yksinkertainen tekstikäyttöliittymä
  - 1-2p monimutkainen tekstikäyttöliittymä
  - 2-3p yksinkertainen graafinen käyttöliittymä
  - 4p laaja graafinen käyttöliittymä
- tiedon pysyväistalletus 4p
  - 0p ei pysyväistalletusta
  - 1-2p tiedosto
  - 3-4p tietokanta
  - 3-4p internet

Harjoitustyön vaikutus kurssipisteisiin

- käyttöliittymä 4p
  - 0p yksinkertainen tekstikäyttöliittymä
  - 1-2p monimutkainen tekstikäyttöliittymä
  - 2-3p yksinkertainen graafinen käyttöliittymä
  - 4p laaja graafinen käyttöliittymä
- tiedon pysyväistalletus 4p
  - 0p ei pysyväistalletusta
  - 1-2p tiedosto
  - 3-4p tietokanta
  - 3-4p internet
- sovelluslogiikan kompleksisuus 3p
- ohjelman lajuus 4p

Harjoitustyön vaikutus kurssipisteisiin

- käyttöliittymä 4p
  - 0p yksinkertainen tekstikäyttöliittymä
  - 1-2p monimutkainen tekstikäyttöliittymä
  - 2-3p yksinkertainen graafinen käyttöliittymä
  - 4p laaja graafinen käyttöliittymä
- tiedon pysyväistalletus 4p
  - 0p ei pysyväistalletusta
  - 1-2p tiedosto
  - 3-4p tietokanta
  - 3-4p internet
- sovelluslogiikan kompleksisuus 3p
- ohjelman lajuus 4p
- ulkoisten kirjastojen hyödyntäminen 1p
- release / suorituskelpoinen jar-tiedosto 1p
- koodin laatu 5p
- virheiden käsittely 2p

Harjoitustyön toimivuus

- Koneiden konfiguraatioissa on eroja, ja tällä kurssilla *ei riitä* että harjoitustyössä tekemäsi sovellus toimii vain omalla koneellasi

Harjoitustyön toimivuus

- Koneiden konfiguraatioissa on eroja, ja tällä kurssilla *ei riitä* että harjoitustyössä tekemäsi sovellus toimii vain omalla koneellasi
- Harjoitustyösi pitää pystyä joka viikko suorittamaan, kääntämään ja testaamaan komentoriviltä käsin laitoksen linux-koneilla (tai uusimmat päivitykset sisältävällä cubbli-linuxilla)
  - muussa tapauksessa työtä ei tarkasteta ja menetät viikon/loppupalautuksen pisteet

Harjoitustyön toimivuus

- Koneiden konfiguraatioissa on eroja, ja tällä kurssilla *ei riitä* että harjoitustyössä tekemäsi sovellus toimii vain omalla koneellasi
- Harjoitustyösi pitää pystyä joka viikko suorittamaan, kääntämään ja testaamaan komentoriviltä käsin laitoksen linux-koneilla (tai uusimmat päivitykset sisältävällä cubbli-linuxilla)
  - muussa tapauksessa työtä ei tarkasteta ja menetät viikon/loppupalautuksen pisteet
- Pääset testaamaan ohjelmaasi laitoksen koneella myös kotoa käsin käyttämällä etätyöpöytää