

Demo 1 / 14.9

Demot palautetaan viimeistään maanantaina klo 12:00 mennessä kurssin `NettiDemoWWW:11ä` <https://www.mit.jyu.fi/demowww/ohj1/>. Voit palauttaa osan tai kaikki tehtäväsi etukäteenkin ja täydentää vastauksia määräaikaan mennessä.

Tehtävät

V1. Tee Villestä (ks: <https://trac.cc.jyu.fi/projects/ohj1/wiki/ville>) kohta 1. Johdanto Villen käyttöön. Tee myös Villestä tehtävä 2.1.

1. Tee **komentoriviä** käyttäen tietokoneellesi (tai Agoran mikroluokassa U:-asemallesi) alla olevan mallin mukainen hakemistorakenne Ohjelmointi 1:n demovastauksia varten. Alla sisennykset kuvaavat sitä, mikä on minkäkin alihakemisto. Vinkkejä kurssin Wikistä: <https://trac.cc.jyu.fi/projects/ohj1/wiki/komentorivi>.

```
kurssit
  ohj1
    demot
      demo1
      demo2
```

Sitten kirjoita komento

```
dir \kurssit /s (Linuxissa ls -R HAKEMISTO)
```

Pitäisi näkyä mm. seuraavia rivejä jos tehtävä on tehty oikein:

```
Directory of U:\kurssit
Directory of U:\kurssit\ohj1
Directory of U:\kurssit\ohj1\demot
Directory of U:\kurssit\ohj1\demot\demo1
Directory of U:\kurssit\ohj1\demot\demo2
```

Tehtävän vastaukseksi luodaan palautettava tiedosto `teht1.txt` seuraavasti:

```
cd \kurssit\ohj1\demot\demo1
dir \kurssit /s >teht1.txt
```

Katso mitä tuli tiedostoon `teht1.txt`:

```
type teht1.txt
```

Pääteohjauksessa ohjaaja tarkistaa, että hakemistot ovat OK. Siellä on viimeinen hetki kysyä, miten hakemistoja tai tekstitiedostoja tehdään! (En uskalla olettaa asiaa esitietona, mutta tiedoston ja hakemiston olemus on ehdottomasti Ohjelmointi 1 -kurssin alussa opittavaa sisältöä!)

2. **Editorin käyttötaito:** Ota sekuntikellolla aika, kauanko juuri sinulla menee alla olevien ohjeiden mukaisen tekstitiedoston kirjoittamiseen. Ei saa huijata! Tämä on leikkimielinen mutta sitäkin vakavampi kilpailu.

Lue ensin tehtävä huolellisesti. Kun olet ensin sisäistänyt kaikki ohjeet, aloita tyhjästä tiedostosta ja käytä ainoastaan valitsemaasi tekstieditoria ja omia käsiäsi. (Tekstieditori on esimerkiksi **ConTEXT**. Mikään Word tai muu "Office"-tyyppinen ohjelma **ei ole tekstieditori**). Kaikki editorin ominaisuudet ovat tietysti käytössä (eli leikkaa-liimaa, etsi-jamuokkaa ja vastaavat kumppanit). **Hiirtä EI SAA** käyttää lainkaan tehtävän aikana. Aloita ajanotto siitä, kun kirjoitat ensimmäisen merkin, ja lopeta siihen kun viimeinen merkki on kirjoitettu. Sisällön tulee olla:

- 1) Alussa eli ihan ensiksi **tasana sata riviä**, joilla jokaisella lukee:
Harjoittelun tekstitiedoston tekemistä!
- 2) Sen jälkeen tasana yksi rivi, jossa on tasana 80 kpl miinusmerkkiä '-' eikä mitään muuta, ei välilyöntejä, ei mitään...
- 3) Sitten neljäkymmentä riviä, joissa on luvut ykkösestä neljäänkymmeneen kahdella numerolla siten että jokaisella rivillä on yksi luku.
Näin alkaisi se pätkä:
01
02
03
... ja näin se päättyisi:
39
40
- 4) Loppuun pitää tulla taas sellainen rivi, jossa on tasana 80 kpl miinusmerkkiä.
Mitään muuta rivejä tai tyhjiä rivejä ei saa olla.

Pysäytä ajanotto. Tallenna nimelle teht2.txt. Jälkeenpäin kirjoita vielä tiedoston loppuun (uudeksi viimeiseksi riviksi heti miinusmerkkirivin perään), kauanko sinulta meni. Ajan tulee olla muotoa mm:ss missä mm on minuuttien määrä ja ss on sekuntien määrä. Tämä perään kirjoita vielä lyhyt kuvaus siitä, mitä ja miten teit tehtävän aikana. Tässä on vastaustiedostosi tähän tehtävään. Muista tallentaa ja nimetä järkevästi. Tulos antaa itsellesi suuntaa nykyisestä tehokkuudestasi tekstin tuottajana suhteessa kurssikavereihin. Esim. Ohjelmointi 1 -kurssin lopussa voi tehdä testin uudelleen, ja kokeilla onko tapahtunut kehitystä.

Mulla meni 01:45, joten tässä ei varmaan kellään mene kovin kauaa. Töpeksin kun tuli 18 kaksi kertaa ja 19 ei yhtään kertaa ja tuo piti korjata. Mutta aika otetaan vasta kun valmis.

3. **Työkaluihin tutustuminen:** Varmista, että sinulla on jossakin käytössäsi minimityökalut Ohjelmointi 1:n suorittamiseen, ja että osaat käyttää niitä:

- Järkevä tekstieditori
- Java 6 SDK (tai uudempi; nimenomaan SDK)
- Komentotulkki
- Internet-yhteys
- SSH-pääteyhteysohjelma (esim. SSH Secure Shell, PuTTY tai vastaava), ohjeet kurssin Wikissä

Agoran mikroluokissa nämä löytyvät valmiina. Kotikoneisiin ne on saatavilla netistä, mutta niiden asentamista ei paljonkaan pystytä henkilökohtaisesti opastamaan. **Kysy kaverilta ja/tai opiskele!** Myös [kurssin Wiki](#) tai postilista ohj01s09p@korppi.jyu.fi voi auttaa.

Kotoa käsin toimiminen **edellyttää pidemmän päälle Java SDK:n asentamista!** Ihan

alkuun voi hätätilassa tulla toimeen pelkästään SSH-päätelysohjelmalla, jos editoi tiedostojaan suoraan yliopiston palvelinkoneella vaikkapa helppokäyttöisellä (mutta erittäin rajoittuneella) tekstieditorilla nimeltä nano.

Kokeile käytännössä ([2. Ensimmäinen Java-ohjelma](#)):

- Tallenna esimerkkiohjelma [Hello.java](#) netistä demohakemistoosi
- Komentotulkkia käyttäen käännä ohjelma ja aja se. Pikaohjeet löytyvät kurssin [Wikistä](#). Tässä lyhyesti:


```
javac Hello.java
java Hello
```

Tehtävän vastauksena palauta tiedosto Hello.java, jonka alkuun olet lisännyt oman nimesi (tai kaikkien ryhmän jäsenten nimet) [kommentteihin](#).

4. **M: 25. ASCII-koodi:** Voi voi. Juhalta on mennyt USB-tikku rikki. Tikulla on vain pieniä tiedonsirpaleita eikä mitään tietoa, mikä sirpale kuuluu mihinkin. Siellä oli paljon MP3-tiedostoja ja kaikkea turhaa... Juha tietää, että yhdessä elintärkeässä tiedostossa hänellä luki selväkielinen teksti isoilla kirjaimilla. Alla oleva bittijono näyttää lupaavalta, mutta mikä teksti siihen on tallennettu:

Taulukko 1:

01000111	01010010	01000001	01000100
01010101	01001110	00100000	01001010
01001111	01001000	01000100	01000001
01001110	01010100	01001001	

Helpotetaan; seuraavassa on samat (taulukon 1) tavun mittaiset jonot [10-järjestelmän](#) lukuina (saatu luettua taulukosta 3 eli 01000111 = 71 jne...), ja malliksi on yksi luku muutettu ymmärrettäväksi kirjaimeksi alla olevaan taulukko 2:een:

Taulukko 2:

71	82	65	68	85	78	...	täydennä vastaavuudet loppuun saakka ...
							=A

Tulkitse kaikki muutkin merkit. Mikä teksti tuohon kohtaan USB-tikun muistia on tallentunut? Montako millimetriä Juha on painanut ohi yhdestä näppäimestä? Hänellä on samanlainen näppäimistö kuin sinulla. Palauta tiedosto teht4.txt jossa alussa nimesi, sitten täydennetty taulukko 2 kirjainten vastaavuuksista ja hutilyönnin millimäärä.

Juha tietää, että hänen käyttämässään tiedostomuodossa merkeille on sovittu seuraavat koodit (pala [ASCII](#)-koodia):

Taulukko 3:

Desimaaliluku	Binäärilukuna	Vastaa merkkiä
32	00100000	[välilyönti]
33	00100001	!
46	00101110	.
65	01000001	A
66	01000010	B
67	01000011	C
68	01000100	D
69	01000101	E

70	01000110	F
71	01000111	G
72	01001000	H
73	01001001	I
74	01001010	J
75	01001011	K
76	01001100	L
77	01001101	M
78	01001110	N
79	01001111	O
80	01010000	P
81	01010001	Q
82	01010010	R
83	01010011	S
84	01010100	T
85	01010101	U
86	01010110	V
87	01010111	W
88	01011000	X
89	01011001	Y
90	01011010	Z

5. **M: 2. Ensimmäinen Java-ohjelma:** Kopioi aluksi 3-tehtävän tiedosto Hello.java tiedostoksi Mina.java:

```
copy Hello.java Mina.java (Linux: cp Hello.java Mina.java)
```

Avaa Mina.java editoriin (esim: `ct Mina.java`) ja muuta luokan nimi Hello nimeksi Mina. Sitten muuta ohjelma sellaiseksi, että se tulostaa sinun nimesi, kännykkäsi valmistajan ja operaattorisi nimen, kunkin eri riveille. Käännä ja aja ohjelma. Palautetaan Mina.java

6. **M: 4. Yksinkertainen graafinen Java-ohjelma:** Tämä tehtävä on tehtävä aidossa työasemassa (valitettavasti ei toimi SSH-päätellä esim. Jalava-koneesta). Kopioi ensi itsellesi kurssin Wiki-sivujen: <https://trac.cc.jyu.fi/projects/ohj1/wiki/graphics> ohjeiden mukaan Graphics.jar-kirjasto. Tee ja aja mallin SimpleGraphics mukainen ohjelma joka piirtää ympyrän ja viivan. Muuta ohjelma sellaiseksi, että se piirtää neliön viivojen avulla nykyisen ympyrän ympärille (mieluiten pienimmän mahdollisen neliön). Tallenna nimelle Neliö.java ja palauta tämä tiedosto.

- B1. Ota ja asenna Alice Wiki-sivujen <https://trac.cc.jyu.fi/projects/ohj1/wiki/Alice> ohjeiden mukaan. Käy ensimmäinen (luistelu) tutoriaali läpi. Palauta tiedosto Teht1b.txt johon kirjoitat parilla rivillä mitä opit ja miltä Alice tuntui, kannattaako siitä ottaa jatkossa lisäesimerkkejä.

GURU-tehtävät

- G1-2 Tee Java -ohjelma, joka tulostaa kaikki ne kokonaisluvut ≤ 1000 , joiden neliöjuuri on kokonaisluku. TDD: Kirjoita ennen koodaamista käsin lukuja ja niiden neliöjuuria.

Tehtävien pisteistä

Jokaisella tehtäväkerralla voi olla Ville-tehtäviä (V=Ville), tavallisia tehtäviä, lisätehtäviä (B=bonus) ja harrastajien tehtäviä (G=Guru).

Tehtävien maksimina pidetään 6/demo (prosentteja laskiessa). Bonus/GURU-tehtävät on tarkoitettu erityisesti siihen, että voi kerätä pisteitä varastoon. GURU-tehtävien tarkoitus on myös tarjota asiaa jo osaaville hieman haastavampia tehtäviä. Kuitenkin kultakin demokerralta lasketaan max. 8 p. Eli vaikka merkitsisi nyt kaikki (=10), siirtyy automaattisesti korkeintaan 8 tehtävää Korppiin.

Vanhat vastaukset tai kavereiden vastaukset

Tehtävistä saattaa liikkua edellisten vuosien vastauksia ja jos niiden tai kavereiden vastauksien käyttämisestä jää kiinni, ovat rangaistukset luntaamiseen verrattavia! Tärkeintä kurssilla on OPPIMINEN, ei demopisteiden kalastelu tai kopiointi. Kopiot saa jo demoissa valmiiksi painetussa muodossa.

Ryhmätyö on sallittua, mutta tämä ilmaistaan laittamalla aina tiedoksi - esimerkiksi ohjelman kommentteihin - kaikkien ryhmään osallistuneiden nimet.

Pääteohjaukset

Tarkista pääteohjausten paikka ja aika aina [Korpista](#)

Tehtävät löytyvät kurssin Wikistä: <https://trac.cc.jyu.fi/projects/ohj1/wiki/paate>

WWW-sivut

Muista kurssin WWW-sivut:

<http://www.mit.jyu.fi/vesal/kurssit/ohjelmointi1/2009/>