

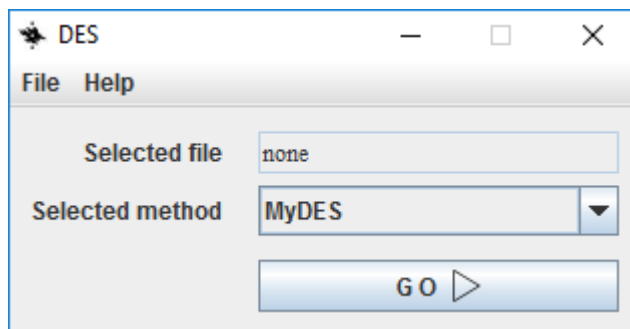
Käyttöohje

Aineopintojen harjoitustyö: Tietorakenteet ja algoritmit
2016 (Periodi II)

Markus Auvo

1 Ohjelman suorittaminen ja toiminnallisuudet

Ohjelma käynnistetään ajamalla *.jar*-tiedosto tai vaihtoehtoisesti käynnistämällä projekti suoraan NetBeans-sovelluksessa. Käynnistyksen jälkeen näytölle tulostuu yksinkertainen graafinen käyttöliittymä. Käyttöliittymässä käyttäjä valitsee sovelluksessa käsiteltävän kuvatiedoston sekä DES-implemентаation. Implementaatioita ovat Javan omilla työkaluilla toteutettu implementaatio *JavaDES* sekä tässä harjoitustyössä toteutettu oma implementaatio *MyDES*. Valinnat tehtyään käyttäjä käynnistää sovelluksen suorituksen painamalla *GO-painiketta*.



Kuva 1: Käyttöliittymä eli sovellusikkuna.

2 Toiminnallisuudet

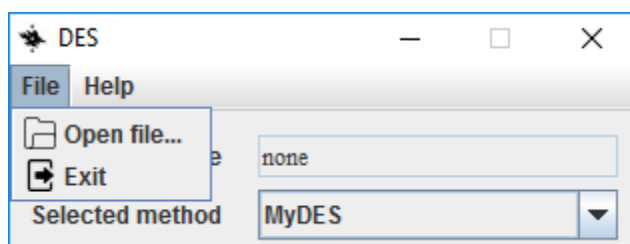
2.1 Valikot

Käyttöliittymän valikkopalkissa on kaksi toimintovalikkoa *File* ja *Help*, joihin voi sovellusta laajennettaessa lisätä toimintoja tarpeen mukaan. Tässä harjoitustyössä pelkästään sovelluksen ydintoiminnot on toteutettu.

2.1.1 File

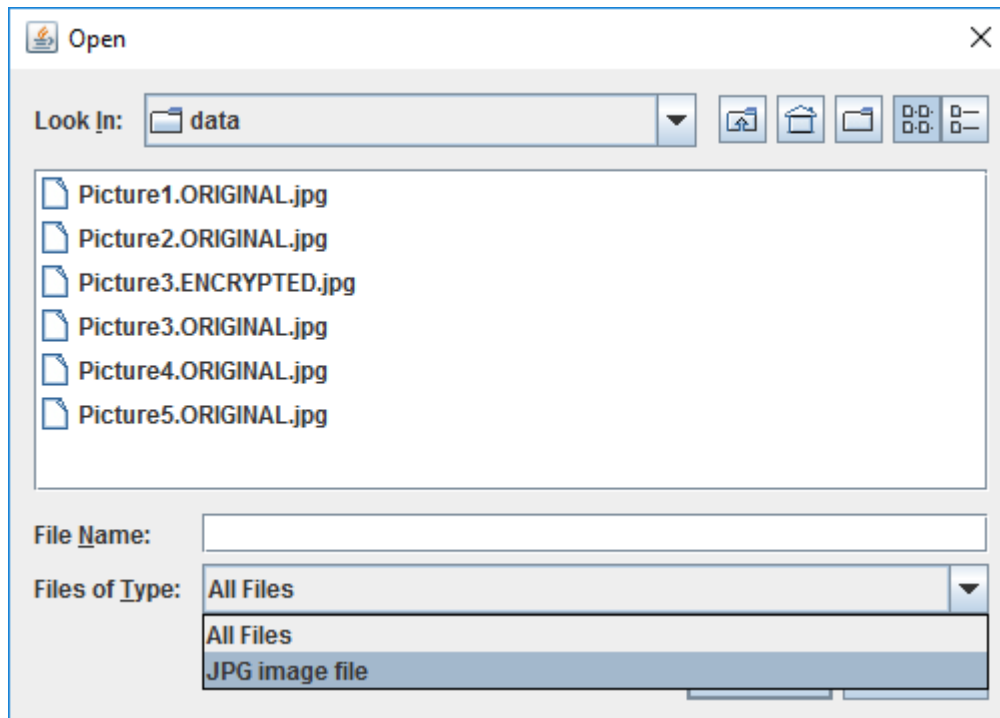
File-valikko koostuu kahdesta toiminnosta:

- Open file...
- Exit.



Kuva 2: File-valikko.

Open file... –toiminto avaa erillisen tiedostonvalintaikkunan, josta käyttäjä voi valita sovelluksessa käsiteltävän kuvatiedoston.



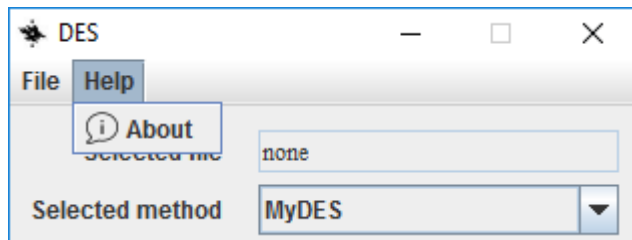
Kuva 3: Tiedostonvalintaikkuna.

Tiedostonvalintaikkuna pyrkii näyttämään oletuksena sovelluksessa oletuksena määritellyn *data*-kansion sisällön, mutta salattavan kuvatiedoston voi valita mistä tahansa kansioista. Sekä salattu että purettu tiedosto luodaan ja tallennetaan samaan kansioon, jossa alkuperäinen kuvatiedosto sijaitsee.

Exit-toiminto sammuttaa sovelluksen samalla tavalla kuin sovelluksen oikeassa yläkulmassa oleva rastipainike.

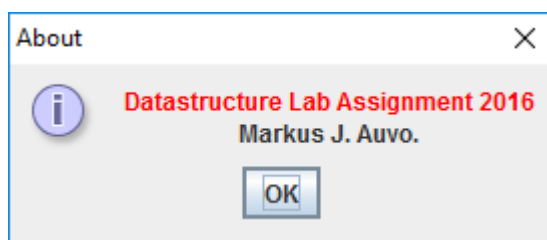
2.1.2 Help

Help-valikko koostuu yhdestä *About*-nimisestä toiminnosta.



Kuva 4: Help-valikko.

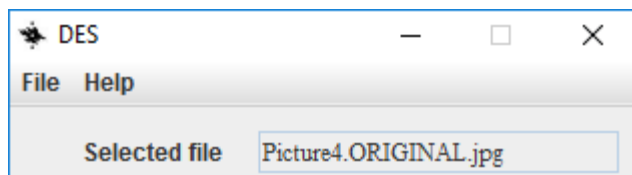
About-toiminto avaa näytölle erillisen informaatioikkunan, josta ilmenee sovellukseen liittyviä tietoja.



Kuva 5: Sovelluksen informaatioikkuna.

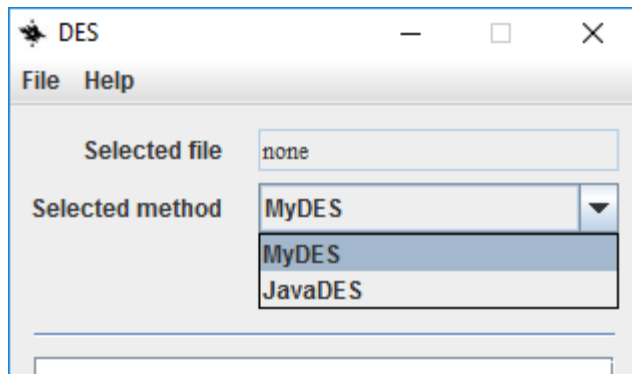
2.2 Toiminnot

Kun käyttäjä on valinnut sovelluksessa käsiteltävän tiedoston, tiedoston nimi tulostuu valitun tiedoston nimi tulostuu käyttöliittymässä olevaan *Selected file* –tekstikenttään.



Kuva 6: Selected file –tekstikenttä.

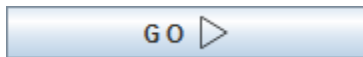
Käyttäjä voi valita käsitelläkö kuvatiedosto Javan omilla salaustyökaluilla varustetulla implementaatiolla *JavaDES* vai tässä harjoitustyössä toteutetulla, omalla implementaatiolla *MyDES*. Valinta tehdään *Selected method* –alasvetovalikosta.



Kuva 7: Selected method –alasvetovalikko.

Oletuksena sovellus käyttää itsetoteutettua implemetaatiota *MyDES*.

Kun käsiteltävä kuvatiedosto ja DES-implemetaatio on valittu, suoritus käynnistetään GO-painikkeella.



Kuva 8: Sovelluksen suorituksen käynnistävä GO-painike.

2.3 Suorittaminen

Kun käyttäjä on aloittanut sovelluksen suorituksen, sovellus sekä salaa käsiteltävän kuvatiedoston että purkaa luomansa salauksen käyttämällä valittua DES-implemetaatiota. Sovelluksen käsittelyn tuloksena syntyneet tiedostot tallennetaan samaan kansioon, josta käsiteltävät tiedostot valitaan. Käsittelyn tuloksena syntyneet tiedostot nimetään seuraavan rakenteen mukaisesti.

$\langle \text{nimi} \rangle . \text{ENCRYPTED} . \langle \text{implemetaatio} \rangle \text{Way.jpg}$
 $\langle \text{nimi} \rangle . \text{DECRYPTED} . \langle \text{implemetaatio} \rangle \text{Way.jpg}$

Esimerkiksi, jos käsiteltäväksi tiedostoksi on valittu Picture4.ORIGINAL.jpg ja tiedosto käsitellään *JavaDES*-implemetaatiolla, sovellus luo ja tallentaa *data*-kansioon seuraavat tiedostot:

- Picture4.ENCRYPTD.JavaWay.jpg
- Picture4.DECRYPTD.JavaWay.jpg

Jos implemetaatioksi olisi valittu itsetoteutettu *MyDES*, tiedostojen nimeksi muodostuisi:

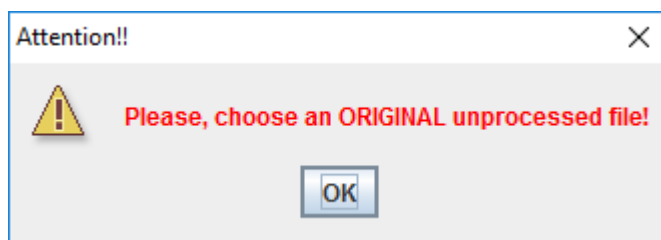
- Picture4.ENCRYPTD.MyWay.jpg
- Picture4.DECRYPTD.MyWay.jpg

3 Suorituksessa käytettävät syötteet

Ohjelman suorituksessa käytetään syötteinä erikokoisia *.jpg*-kuvatiedostoja. Sovelluksen käsiteltäväksi annettavat kuvatiedostot tulee nimetä seuraavaa rakennetta käyttäen:

<nimi>.ORIGINAL.jpg

Tässä harjoitustyössä tälle käytännölle on syynä, että harjoitustyön tuloksena syntyneitä tiedostoja voidaan tällöin halutessa vertailla alkuperäisten tiedostojen kanssa. Mikäli tiedostoa ei ole nimetty edellä mainitun nimeämiskäytännön mukaisesti, käyttäjä saa siitä ilmoituksen.



Kuva 9: Ilmoitus, kun tiedosto on nimetty nimeämiskäytännön vastaisesti.

Alkuperäiset kuvatiedostot voi sijoittaa mihin tahansa kansioon, mutta sovellus hakee kuvatiedostot oletuksena *data*-kansioista, jos sellainen on olemassa.

4 Hakemistot ja tiedostot

Sovelluksen hakemistorakenne on jakautuu kahteen osaan, sovelluskansioon *app* ja dokumentaatiokansioon *docs*. Sovelluksen hakemistopuu seuraavanlainen:

```
<sovelluskansio>/
  app/
    data/
    resources/
    src/
      Datastructure_lab_2016/
        algo/
        util/
        view/
    test/
      Datastructure_lab_2016/
        algo/
        util/
  docs/
```

Tässä harjoitustyössä hakenistorakenteesta on jätetty pois NetBeans-projektille spesifiset hakemistot sekä tiedostot ja esitelty ainoastaan itse sovelluksen kannalta oleelliset hakemistot. Lähdekoodihakemisto `src` ja yksikkötestihakemisto `test` sisältävät saman hakemistorakenteen, kuten ylläolevasta hakemistopuusta käy ilmi. Yksikkötestitiedostot on nimetty samalla tavalla kuin lähdekooditiedostot, mutta luokan nimen perään on lisätty sana *Test*. Esimerkiksi tiedostojen käsittelyyn luotua työkaluluokkaa *FileTool* vastaa yksikkötestiluokka *FileToolTest*.

Seuraavassa on esitelty sovelluksen kannalta oleelliset kansiot.

data/

Kansioon on harjoitustyön demonstraatiota varten oletuksena sijoitettu sovelluksen käsittelemät alkuperäistiedostot sekä sovelluksen luomat tiedostot.

resources/

Kansio sisältää tässä harjoitustyössä pääasiassa käyttöliittymän käyttämiä resursseja, kuten kuvaketiedostoja.

src/

Kansio sisältää sovelluksen lähdekooditiedostot

Datastructure_lab_2016

Sovelluksen pääkansio ja sisältää sovelluksen pääluokan *Main*.

algo/

Kansio sisältää DES-implementaatioluokat *JavaDES* ja *MyDES*.

util/

Kansio sisältää sovelluksen kannalta hyödyllisiä työkaluluokkia, kuten *FileTool*, *ArrayTool*, yms.

view/

Kansio sisältää sovelluksen näkymän eli käyttöliittymän, *GUI*.

test/

Kansio sisältää sovelluksen yksikkötestiluokat DES-implementaatioluokille ja työkaluluokille.

Datastructure_lab_2016

Sovelluksen päätestikansio ja sisältää sovelluksen päätestiluokan *MainTest*. Kansio sisältää yksikkötestiluokat *algo*- ja *util*-kansioissa oleville luokille.