Ką aš padariau...

- Duomenų skaitymas iš duomenų failo (lokalaus disko)
- Duomenų skaitymas iš atminties (clipboard)
- Smulkūs pataisymai.

Tai naudinga, tačiau neaišku kodėl tobulinot archyvinę versiją
GIILIOTINA-2,
vietoj pagrindinės
GIILIOTINA-GMJ2,
kurią nagrinėjom per paskaitą ir laboratorinius.
Manau, kad bus nesunku perkelti tai ką padarėt į pagrindinę
versija nes būtent ten vykdomas euristikų optimizavimas.
Kad būtų aiškiau prisegu pagrindinės versijos archyvą 'decomp.tgz'
kur nauji tik du java failai 'Cut.java' ir 'Domain.java' kataloge 'tasks' kurie 'pririša' archyvinę versiją prie optimizavimo metodų esančių archyve 'gmj20.rym.jar'. Giljotinos failai yra kataloge 'analysis', jie labai panašūs skirasi tik vienas pavadinimas: tai kas pas jus 'Cut.java' kataloge 'analysis' vadinama 'cutAnalyser.java', trūksta tik jūsų i/o failų 'inputCB.java' ir 'inputFile.java'.

Smulkūs pataisymai:

- Įdėtas pagrindinis menių, iš kurio pasirenkamas duomenų nuskaitymo tipas.
- Sukeistas Width ir Height parametrų perdavimas *worker* klasėj (anksčiau buvo atvikščiai);

Gerai

Iškilę nesklandumai.

- Buvo tik vienas, LABAI didelis iškilęs nesklandumas, kurį aiškinausi didžiąją laiko dalį. Iki šiol nesuprantu, kodėl sukompiliavus *.java failus su raktu –d, kuris nurodo į kurią direktoriją surašyti *.class failus, naršyklė nepastebi, kad sudarytas archyvas yra užregistruotas su jarsigneriu, nors patikrinus su komanda *jarsigner –verify Cut.jar* matydavau, kad archyvas yra užregistruotas, tačiau naršyklė to nesuprasdavo... Ties šia bėda užsisėdėjau gal 5+ valandos, kol radau sprendimą,

- kad negalima naudoti –d rakto kompiliuojant... Tikra to žodžio prasme, sugaišau visą dieną...

Pridėtas labai elementarus skriptukas 'cmp-jar.sh' kurį naudojau kompiliavimui-archyvavimui kuriame nereikia nurodyti '-d'.
Įdomu kokios naršyklės 'nepasirašo' naudojant '-d' ?