Universiteti i Prishtinës
Fakulteti i Inxhinierisë Elektrike dhe Kompjuterike

Projekti në lëndën Siguria e të Dhënave

Faza e parë, Grupi 13

Blerim Rexha, Arbnor Halili, Edon Gashi

Mars 2020

Abstrakt: Kjo fazë ka për qëllim familjarizimin tuaj me teknikat klasike të enkriptimit dhe përpunimin e të dhënave tekstuale. Këto teknika do t'i demonstroni duke e shkruar një program të thjeshtë i cili ofron disa shërbime përmes komandave. Ju lusim ta lexoni me kujdes këtë dokument dhe të veproni sipas udhëzimeve.

Përmbajtja

Kërkesat

Detyra juaj është ta shkruani një console program me emrin `ds` i cili pranon komanda përmes argumenteve. Programi juaj do t'i analizojë argumentet, dhe varësisht nga përmbajtja e tyre do ta ekzekutojë ndonjërën prej komandave të specifikuara.

Argumentet janë termet që vijnë pas programit. Shumica e ambienteve programuese mundësojnë leximin e tyre, psh. `String[] args` në Java, `process.argv` në Node, `sys.argv` në Python, etj.

Në rast se argumentet mungojnë ose janë të jo-valide, atëherë do ta shfaqni një tekst me udhëzime rreth përdorimit dhe do ta mbyllni programin me kod dalës (exit code) 1. Poashtu, nëse gjatë ekzekutimit ka ndonjë dështim për shkak të hyrjeve jo-valide ose ndonjë gabimi gjatë shkrim-leximit në fajlla, programi duhet ta trajtojë gabimin dhe ta shfaqë në ekran një mesazh përshkrues.

Nuk ka kufizim sa i përket gjuhës programuese, por duhet ta keni parasysh që:

- Kërkesat të plotësohen ashtu siç janë specifikuar.
- Të gjitha veprimet të kryhen nga programi i njëjtë, pra jo nga një program për secilën komandë.
- Kodet që i merrni të gatshme nga interneti duhet të referencohen në komente.

Në vazhdim e gjeni specifikimin e komandave, ku përfshihen edhe shembuj të përdorimit të tyre.

Shënim: Edhe pse preferohet të implementohet programi në console, jeni të lirë ta zhvilloni në platformë tjetër sipas dëshirës (GUI, Web) me kushtin që të ofrohet funksionaliteti ekuivalent dhe të respektohen rregullat e mësipërme.

Komanda beale

Beale Cipher zëvendëson secilën shkronjë të plaintextit me pozitën e asaj shkronje në një libër. Lexo më shumë ***këtu***.

Në shembujt e mëposhtëm supozojmë që kemi fajllin `libri.txt` me përmbajtjen:

```
the quick brown fox jumps over the lazy dog
```

Nënkomanda encrypt

Sintaksa: `ds beale encrypt <book> <plaintext>`

Enkripton plaintextin `<plaintext>` duke u bazuar në një text file `<book>` që e paraqet librin. Ciphertexti i fituar shfaqet në ekran.

Shembull:

```
$ ds beale encrypt libri.txt "pershendetje"
24 29 12 25 33 34 15 41 3 32 21 3
```

Nënkomanda decrypt

Sintaksa: `ds beale decrypt <book> <ciphertext>`

Dekripton ciphertextin `<ciphertext>` duke u bazuar në një text file `<book>` që e paraqet librin. Plaintexti i fituar shfaqet në ekran.

Shembull:

```
$ ds beale decrypt libri.txt "24 29 12 25 33 34 15 41 3 32 21 3"
pershendetje
```

Komanda four-square

Cipher që transformon plaintextin përmes 4 tabelave 5×5 dhe disa rregullave. Lexo më shumë ***këtu***.

Nënkomanda encrypt

Sintaksa: `ds four-square encrypt <key1> <key2> <plaintext>`

Enkripton plaintextin `<plaintext>` me çelësat `<key1>` dhe `<key2>` përmes Four-square Ciper dhe e shtyp ciphertextin në ekran.

Shembull:

```
$ ds four-square encrypt siguria dhenave "takohemi neser"
nndoeelbmetepw
```

Nënkomanda decrypt

Sintaksa: `ds four-square decrypt <key1> <key2> <ciphertext>`

Dekripton ciphertextin `<ciphertext>` me çelësat `<key1>` dhe `<key2>` përmes Four-square Cipher dhe e shtyp plaintextin në ekran.

Shembull:

```
$ ds four-square decrypt siguria dhenave "nndoeelbmetepw"
takohemineser
```

Komanda case

Sintaksa: `ds case <case> <text>`

E konverton tekstin `<text>` në madhësinë e dhënë `<case>`, e cila mund të jetë: `lower`, `upper`, `capitalize`, `inverse`, `alternating`.

Shembull:

```
$ ds case lower "Pershendetje nga FIEK!"
pershendetje nga fiek!

$ ds case upper "Pershendetje nga FIEK!"
PERSHENDETJE NGA FIEK!

$ ds case capitalize "Pershendetje nga FIEK!"
Pershendetje Nga Fiek!

$ ds case inverse "Pershendetje nga FIEK!"
pERSHENDETJE NGA fiek!

$ ds case alternating "Pershendetje nga FIEK!"
pErShEnDeTjE NgA FiEk!
```

Pikë shtesë

Të shtohet varianti për **case** me vlerën `sentence`:

```
pershendetje, Fjalia E pare. FJALIA E DYTE! fjAlia E trEte.
Pershendetje, fjalia e pare. Fjalia e dyte! Fjalia e trete.
```

Vlerësimi

Kjo fazë vlerësohet me maksimalisht 10 pikë.

Ju do të gjykoheni në bazë të:

- Kërkesave të plotësuara.
- Cilësisë së kodit.
- Korrektësisë në menaxhimin e repository.
- Njohurive teorike.
- Njohurive teknike.

Pikët shtesë: Në disa vende mund të jenë cekur kërkesa shtesë. Nuk jeni të obliguar t'i plotësoni këto kërkesa, por do të shpërbleheni me pikë shtesë nëse arrini t'i implementoni.

Dorëzimi

Kodi burimor i programit duhet të vendoset në GitHub para datës **20.03.2020 23:59**. Në rast se repository bëhet privat atëherë duhet të ofrohet qasje leximi në llogarinë arbnorhalili (arbnor.halili@uni-pr.edu).

Në `README` duhet ta bëni një përshkrim të shkurtër të projektit. Pikat që duhet t'i diskutoni janë:

1. Udhëzimet për ekzekutimin e programit.
2. Përshkrim i shkurtër për secilën komandë.
3. Rezultatet e ekzekutimit me nga një shembull për secilën komandë dhe nënkomandë.

Gjithashtu kujdesuni ta vendosni një `.gitignore` adekuate ashtu që mos të ngarkohen fajlla të padobishëm në repository.

Kujdes: Cilido lloj i plagjiaturës, qoftë në kod apo në përshkrim, do të ndëshkohet me **0 pikë për të gjitha grupet ku gjendet materiali i kopjuar**, pavarësisht se cili grup e ka punuar i pari.