



Diplomski studij

**Informacijska i komunikacijska
tehnologija**

Telekomunikacije i informatika

Računarstvo

Programsko inženjerstvo i
informacijski sustavi

Računarska znanost

Raspodijeljena obrada velike količine podataka

1. Domaća zadaća

Ak. g. 2017./2018.

Zadatak 1: Pristup HDFS-u preko komandne linije

Cilj zadatka jest uspješno pokrenuti raspodijeljeni datotečni sustav *HDFS* na platformi *Hadoop* te obaviti neke osnovne datotečne operacije. Dvije su mogućnosti pokretanja platforme Hadoop: 1) pokretanje unaprijed pripremljenog virtualnog stroja s platformom Hadoop u pseudo-raspodijeljenom načinu rada i 2) pokretanje samostalno instalirane platforme Hadoop. Prvi način pokretanja se preporuča zbog jednostavnosti, a opisan je u zasebnom dokumentu koji se može pronaći na web stranici predmeta. Drugi način pokretanja ovisi o inačici operacijskog sustava i puno je složeniji (posebice za Windows). Nastavnici i asistenti će pružiti pomoć samo za instalaciju i pokretanje platforme Hadoop na virtualnom stroju. Ukoliko se netko od studenata ipak odluči za drugi način, tada upute za instalaciju i pokretanje platforme Hadoop u pseudo-raspodijeljenom načinu rada na različitim operacijskim sustavima može pronaći na sljedećim poveznicama:

- [Instalacija za operacijski sustav Linux](#)
- [Instalacija za operacijski sustav Windows](#)

Uspješnim rješenjem ovog zadatka steći ćete sljedeća znanja:

- **osnovni rad s datotečnim sustavom HDFS-u korištenjem komandne linije**

Navedene korake prođite iterativno i prijedite na idući korak tek nakon uspješne izvedbe prethodnog.

IZVJEŠTAJ: Na sustav Moodle predajete sve naredbe koje upisujete za izvođenje pojedinog koraka (osim koraka 1) te eventualne odgovore na postavljena pitanja.

1. U svojoj korisničkoj mapi (tj. u mapi ~ na Linuxu) stvorite podmapu ROVKP_DZ1 i postavite ju kao radnu. Sve naredbe daljnjih koraka izvoditi ćete iz ove radne mape.
2. Pokrenite HDFS pomoću skripte `start-dfs.sh`.
3. Prikažite sadržaj mape `/user/rovkp` na HDFS-u.
4. Uz pomoć naredbe `wget` dohvatite datoteku `gutenberg.zip` (veličine 151 MB) sa sljedeće poveznice:

<http://svn.tel.fer.hr/gutenberg.zip>

Ova datoteka sadrži uzorak knjiga Projekta Gutenberg, volonterskog pothvata koji se bavi digitalizacijom knjiga radi očuvanja kulturnih dobara čovječanstva. Ovaj uzorak predstavlja kolekciju od 596 knjiga na engleskom jeziku u formatu TXT, a originalno je izdan 2003. kao CD uzorak od 600 tada najpopularnijih knjiga Projekta. Knjige su pohranjene u mapama kako su originalno izdane na CD-u, prema tada aktualnom katalogu Projekta Gutenberg.

5. Prenesite navedenu datoteku na HDFS u mapu `/user/rovkp`.
6. Pogledajte dijagnostičke podatke datoteke `gutenberg.zip` na HDFS-u. Odgovorite na sljedeća pitanja:
 - a. Od koliko blokova se sastoji datoteka?
 - b. Koji je njihov replikacijski faktor?
 - c. S obzirom na veličinu datoteke, je li ona prilagođena prirodi HDFS-a?
7. Napravite sigurnosnu kopiju datoteke `gutenberg.zip` na lokalnom datotečnom sustavu grozda.
8. Prenesite datoteku `gutenberg.zip` s HDFS-a na lokalni datotečni sustav grozda.
9. Korištenjem naredbe `md5sum` provjerite je li ova kopija datoteke istovjetna originalnoj verziji.

Zadatak 2: Rad sa datotekama uz pomoć Javinog programskog sučelja

Cilj zadatka je napisati Javin program za spajanje kolekcije knjiga dobivenih u prethodnom zadatku u jedinstvenu tekstualnu datoteku.

Uspješnim rješenjem ovog zadatka steći ćete sljedeća znanja:

- razvoj i izvršavanje programa u razvojnom sučelju NetBeans
- rad sa tokovima podataka lokalnog datotečnog sustava

Za domaću zadaću navedeni zadatak ćete izvesti na vašem računalu, tj. izvor i odredište će biti staza lokalnog datotečnog sustava (nećete trebati koristiti virtualni stroj).¹ Za rad s datotekama koristite standardne klase iz paketa `java.io`. Program ćete napisati i izvršiti uz pomoć razvojne okoline NetBeans. Nastavnici i asistenti neće pomagati pri instalaciji i pokretanju ostalih razvojnih okolina kao što su Eclipse i IntelliJ IDEA. Upute za instalaciju i pokretanje razvojne okoline NetBeans se nalaze na sljedećem linku: <https://netbeans.org/community/releases/82/install.html>

IZVJEŠTAJ: Na sustav Moodle predajete izvorni programski kod te odgovore na pitanja koja slijede iza opisa zadataka.

Detaljni opis zadatka:

Datoteku `gutenberg.zip` iz prethodnog zadatka pohranite na lokalni datotečni sustav vašeg računala (ne virtualnog stroja!). Raspakirajte datoteku u željenu mapu. Pregledajte sadržaj novostvorene mape *gutenberg*. Uočite da se u njoj nalazi izvjestan broj podmapa (naziva `etext00`, `etext01` itd.) koje sadrže niz tekstualnih datoteka (knjiga na engleskom jeziku).

Javin program kojeg trebate napisati mora iterativno proći kroz ovu strukturu i podatke iz svih tekstualnih datoteka pohraniti u novu datoteku `gutenberg_books.txt`. Sve izvorne datoteke moraju se čitati i prenositi redak po redak. Nakon uspješnog izvođenja programa na zaslone se treba ispisati ukupni broj pročitanih redaka.

Nakon uspješnog pokretanja odgovorite na sljedeća pitanja:

- Koja je veličina konačne datoteke `gutenberg_books.txt`?
- Koliko je ukupno redaka pročitano?
- U slučaju kad bi tu datoteku pohranili na HDFS s veličinom blokova 128 MB i faktorom replikacije 3, koliko bi se ukupno blokova stvorilo na HDFS-u?
- Koliko vremena se izvodio program? Kakvo bi bilo očekivano vrijeme izvođenja kada bi se taj program izvršavao na Hadoopovom grozdu, uz pohranu na HDFS?

¹ U sklopu nastupajuće laboratorijske vježbe stvoriti ćete zadatak identične funkcionalnosti, samo će se izvršavati u sklopu Hadoopovog grozda, tj. odredište će umjesto lokalnog datotečnog sustava biti HDFS.

Zadatak 3: Izvršavanje Javinog programa unutar Hadoopovog grozda

Cilj zadatka je stvoriti prototip Javinog programa koji može istovremeno pristupiti i lokalnom i raspodijeljenom datotečnom sustavu (tj. HDFS-u). Program mora instancirati objekte koji predstavljaju lokalni i raspodijeljeni datotečni sustav te provjeriti postojanje odabrane datoteke/mape na svakom od tih datotečnih sustava.

Uspješnim rješenjem ovog zadatka steći ćete sljedeća znanja:

- **priprema okoline NetBeans za razvoj programa koji će se izvršavati na Hadoop grozdu**
- **rad s Hadoop klasama za upravljanje datotekama i datotečnim sustavima**

IZVJEŠTAJ: Na sustav Moodle predajete izvorni programski kod.

Detaljni opis zadatka:

Za razliku od prethodnog zadatka, program koji ćete pisati u sklopu ovog zadatka neće se izvoditi na lokalnom stroju, već ćete ga izvesti unutar Hadoopovog grozda (koji je u pseudo-raspodijeljenom načinu rada). Shodno tome, program će umjesto klasa paketa `java.io` koristiti klase iz paketa `org.apache.hadoop`.

Kako biste pripremili okolinu NetBeans za razvoj ovakvog programa potrebno je napraviti sljedeće:

- stvoriti novi projekt odabirom opcije *New Project -> Maven -> Java Application*
- unutar *Project Explorer-a* otvoriti *Project Files -> pom.xml*
- desni klik unutar prikaza XML dokumenta *Insert Code -> Dependency*
- u polje *Query* upisati "Hadoop client"
 - o napomena: ukoliko nemate otprije indeksirani repozitorij traženje prvi put može potrajati!
- odabrati *org.apache.hadoop: hadoop-client -> 2.6.5 [jar] -central*
- kliknuti *Add*

Na ovaj način vaš će projekt automatski uključiti sve klase potrebne za razvoj Hadoop klijenata, što ćete vidjeti pojavom sljedećeg elementa u *pom.xml* dokumentu:

```
<dependencies>
    <dependency>
        <groupId>org.apache.hadoop</groupId>
        <artifactId>hadoop-client</artifactId>
        <version>2.6.5</version>
    </dependency>
</dependencies>
```

NAPOMENA: Ukoliko u sljedećim koracima okolina NetBeans ne pronalazi klase iz paketa `org.apache.hadoop`, spremite projekt te nakon desnog klika na ime projekta u izborniku odaberite opciju "Clean and Build", a potom pričekajte da se spuste sve potrebne klase.

Potrebno je dodati još nekoliko elemenata ovom XML dokumentu. Kako okolina NetBeans ne nudi mogućnost automatskog dodavanja sljedećih redaka, unesite ih ručno iznad gore navedenog elementa `<dependencies>`:

```

<name>IME_PROJEKTA</name>
<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-jar-plugin</artifactId>
      <version>2.6</version>
      <configuration>
        <archive>
          <manifest>
            <addClasspath>true</addClasspath>
            <mainClass> IME_KLASE </mainClass>
          </manifest>
        </archive>
      </configuration>
    </plugin>
  </plugins>
</build>

```

Za ime projekta stavite ime koje ste odabrali, a za klasu postavite onu koja će implementirati *main* metodu a koju ćete stvoriti u idućim koracima zadatka. Ovime ste načinili sve pripremne korake za uspješno pisanje Javinog programa koji će se izvršavati u Hadoopovom grozdu.

Sada stvorite novu Javinu klasu koja će unutar svoje metode `main` treba napraviti sljedeće:

- instancirati objekt *Configuration*
- instancirati objekt *LocalFileSystem* (reprezentaciju lokalnog datotečnog sustava)
- instancirati objekt *FileSystem* (reprezentaciju distribuiranog datotečnog sustava)
- instancirati dva objekta *Path* (reprezentaciju mape/datoteke lokalnog datotečnog sustava te mape/datoteke distribuiranog datotečnog sustava)
- uz pomoć metoda *isFile(Path)* / *isDirectory(Path)* klase *FileSystem* uspješno provjeriti valjanost definiranih staza do odabranih mapa / datoteka

VAŽNO: ovu klasu nećete izvršavati unutar Netbeans sučelja, već ćete ju zapakirati u JAR datoteku koju ćete prebaciti na klijentski sloj (virtualku) sa instaliranom Hadoop-om, a potom na HDFS. JAR datoteku možete stvoriti odabirom opcije *Run -> Build Main Project* ili jednostavno klikom na tipku F11.

Ako se projekt uspješno preveo, u prozoru statusne konzole potražite redak s imenom JAR datoteke na lokalnom datotečnom sustavu. Sada učinite sljedeće:

- prenesite JAR datoteku na lokalni datotečni sustav Hadoopovog grozda (npr. uz pomoć programa WinSCP ako koristite Windowse).
- izvršite JAR datoteku unutar Hadoopovog grozda naredbom

```
hadoop jar <staza do JAR datoteke> <ime klase 'main'>
```
- ukoliko program ne radi, vratite se u okolinu NetBeans, popravite izvorni kod i ponovite postupak

Rješenjem ovog zadatka uspješno ste riješili prvu domaću zadaću. Čestitamo! ☺