

Задача 1 - вывод данных пула констант

Эта задача для того, чтобы научиться или продемонстрировать вашу возможность писать простые программы разбора бинарных данных и прочувствовать на практике формат java класс файлов.

Функциональные требования

Результат должен быть в виде исполняемой консольной программы следующего формата:

```
./dump_constant_pool <class file>
```

Программа должна выводить в стандартный вывод следующие данные:

```
Class file format version: <version number of class file format>

<entry number> <entry type>: <entry value as string>
```

Для примера формата можно взять формат вывода `javap`. Ниже пул констант простого hello world:

```
package ru.nsu.ojp2020;
public class App {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

```
#1 = Class          #2          // ru/nsu/ojp2020/App
#2 = Utf8           ru/nsu/ojp2020/App
#3 = Class          #4          // java/lang/Object
#4 = Utf8           java/lang/Object
#5 = Utf8           <init>
#6 = Utf8           ()V
#7 = Utf8           Code
#8 = Methodref      #3.#9      // java/lang/Object."<init>":()V
#9 = NameAndType    #5:#6      // "<init>":()V
#10 = Utf8         LineNumberTable
#11 = Utf8          LocalVariableTable
#12 = Utf8          this
#13 = Utf8          Lru/nsu/ojp2020/App;
#14 = Utf8          main
#15 = Utf8          ([Ljava/lang/String;)V
#16 = Fieldref      #17.#19    //
```

```
java/lang/System.out:Ljava/io/PrintStream;
#17 = Class                #18          // java/lang/System
#18 = Utf8                  java/lang/System
#19 = NameAndType           #20:#21      // out:Ljava/io/PrintStream;
#20 = Utf8                  out
#21 = Utf8                  Ljava/io/PrintStream;
#22 = String                #23          // Hello World!
#23 = Utf8                  Hello World!
#24 = Methodref             #25:#27      // java/io/PrintStream.println:
(Ljava/lang/String;)V
#25 = Class                #26          // java/io/PrintStream
#26 = Utf8                  java/io/PrintStream
#27 = NameAndType           #28:#29      // println:(Ljava/lang/String;)V
#28 = Utf8                  println
#29 = Utf8                  (Ljava/lang/String;)V
#30 = Utf8                  args
#31 = Utf8                  [Ljava/lang/String;
#32 = Utf8                  SourceFile
#33 = Utf8                  App.java
```

Рекомендации по реализации

- Языки программирования **C** или **C++** или **Java** (Конкретно для этой задачи идеально подходит **C**)
- Сторонние библиотеки для разбора байткода класс не используются

Ресурсы

- <https://docs.oracle.com/javase/specs/jvms/se15/html/jvms-2.html#jvms-2.5.5>
- <https://docs.oracle.com/javase/specs/jvms/se15/html/jvms-4.html#jvms-4.1>
- <https://docs.oracle.com/javase/specs/jvms/se15/html/jvms-4.html#jvms-4.4>

Формат сдачи задачи

- Отдельный git репозиторий на любом хостинге.
- Наличие README файла с инструкциями как собрать проект и другой информацией, которая может быть полезна

Срок сдачи

25 сентября 2020