

Структура class-файла простейшего приложения на Java

Пример. Простейшее приложение на Java.

файл App.java:

```
package hello;

public class App {

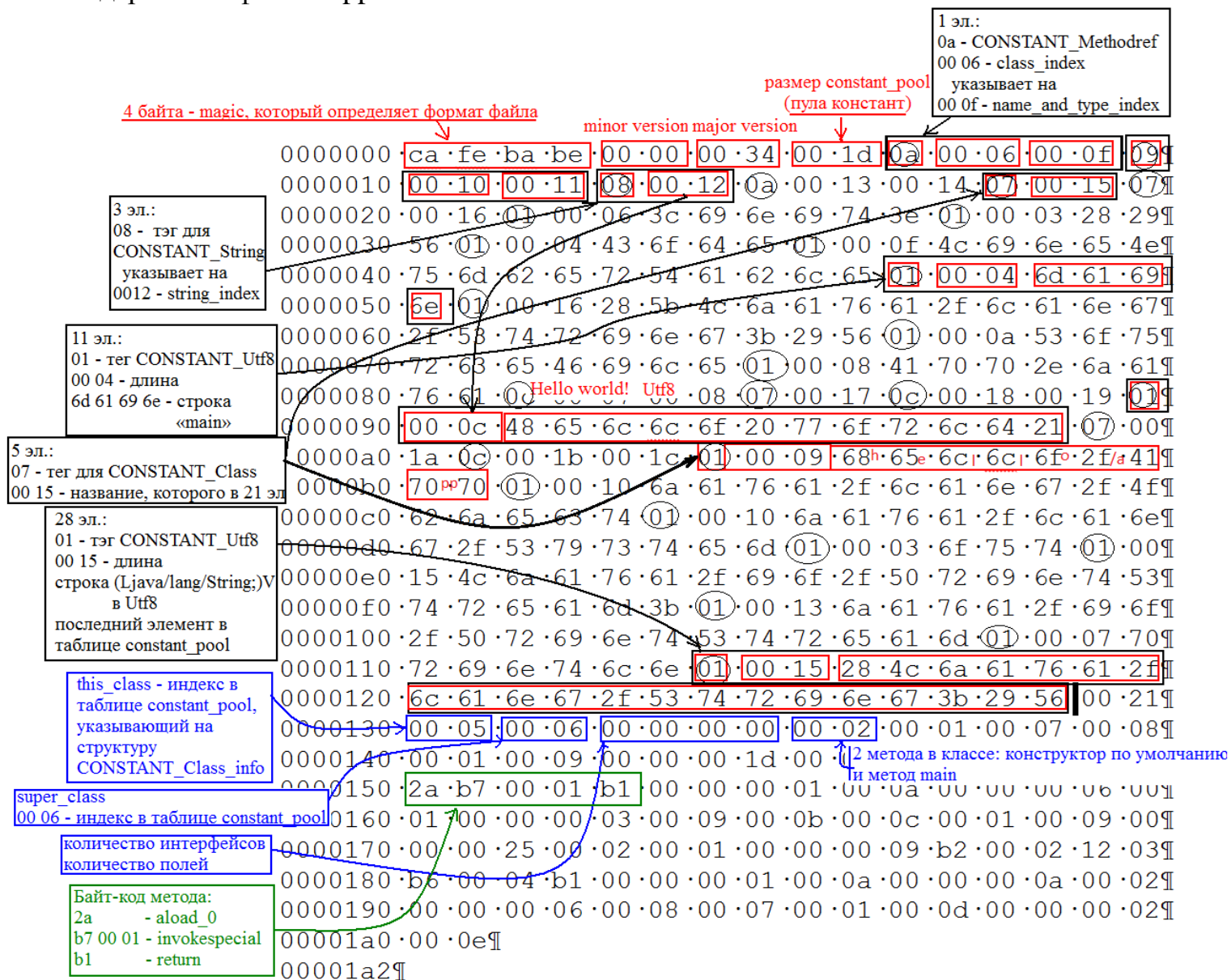
    public static void main(String[] args) {
        System.out.println("Hello world!");
    }

}
```

Скомпилируем файл командой:

```
javac src/hello/App.java -d classes/
```

Содержимое файла App.class:



Обозначения, используемые на рисунке

черный цвет — пул констант:

начало каждого элемента пула обозначено черным кругом;

сам элемент в черной рамке;

поля структуры элементов тега в красных рамках.

синий цвет — общее описание класса и входящих в него методов.

зеленый цвет — байт-код метода.

Файл App.class начинается со стартового (магического) числа: всегда 0xCAFEBABE. Данное число присутствует в каждом классе и является обязательным флагом для JVM. С его помощью система понимает, что перед ней class-файл.

Следующие четыре байта class-файла содержат старший и младший номера версий Java. В примере major версия 0x0034, что соответствует Java SE 8.

С девятого байта начинается пул констант, в котором содержатся все константы класса. Так как в каждом классе их может быть различное количество, то первые два байта указывают на **длину пула констант** (0x001d равен 29 – числу записей в пуле констант + 1 – длина пула констант в нулевом индексе массива), то есть пул констант представляет из себя массив переменной длины. Каждая константа занимает один элемент в этом массиве. Во class-файле константы указываются целочисленным индексом, который обозначает их положение в массиве. Начальная константа имеет индекс 1, вторая константа — 2 и т. д.

Каждый элемент пула констант начинается с однобайтового тега, определяющего его тип. Это позволяет JVM понять, как правильно обработать следующую далее константу. Всего зарезервировано 14 типов констант:

Тип константы	Значение тега	
CONSTANT_Class	7	# класс
CONSTANT_Fieldref	9	# поле
CONSTANT_Methodref	10	# метод
CONSTANT_InterfaceMethodref	11	
CONSTANT_String	8	# строку
CONSTANT_Integer	3	
CONSTANT_Float	4	
CONSTANT_Long	5	
CONSTANT_Double	6	
CONSTANT_NameAndType	12	
CONSTANT_Utf8	1	
CONSTANT_MethodHandle	15	
CONSTANT_MethodType	16	
CONSTANT_InvokeDynamic	18	

где # – ссылка на элемент пула констант

Например, если тег указывает, что константа является строкой, JVM получает значение тега 1 и обрабатывает следующее за тегом 2 байта как длину строки в байтах, которая необходима для получения всех ее символов (см. 11 элемент пула констант на рисунке: 01 (тег), 0004 (длина константы), 6d 61 69 6e (строка `main` в Utf8)).

Первый элемент пула констант:

10 – CONSTANT_Methodref (ссылка на метод)

0x0006 (индекс класса – это ссылка на индекс класса в 6-ом элементе пула констант)

0x000f (индекс 15 указывает на название и тип метода)

2-й элемент:

09 – CONSTANT_Fieldref

0x0010 – индекс класса в 16-м элементе пула

0x0011 – название и тип в 17-м элементе пула

3-й элемент:

08 – CONSTANT_String (ссылка на строковую константу)

0x0012 – ссылка на 18-й элемент пула, где лежит строковая константа "Hello world!"

4-й элемент:

10 – CONSTANT_Methodref (ссылка на метод)

0x0013 – класс которого описан в 19 элементе пула

0x0014 – по ссылке 20 название и тип метода

5-й элемент:

07 – CONSTANT_Class

0x0015 – название класса в 21 элементе пула (строка из 9 символов Utf8: hello/app)

6-й элемент:

07 – CONSTANT_Class

0x0016 – название класса в 22 элементе пула (строка из 10 символов Utf8: java/lang/Object). 1-й элемент пула относится к этому классу.

7-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0006 – длина константы

3c 69 6e 69 74 3e – строковая константа из 6 символов <init> (так помечаются конструкторы)

8-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0003 – длина константы

28 29 56 – строковая константа из 3 символов ()V (описание конструктора без параметров или сигнатура)

9-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0004 – длина константы

43 6f 64 65 – строковая константа из 4 символов Code

10-й элемент:

01 – CONSTANT_Utf8 (константа)

0x000f – длина константы

LineNumberTable – строковая константа из 15 символов

11-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0004 – длина константы

main – строковая константа из 4 символов

12-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0016 – длина константы

([Ljava/lang/String;)V – строковая константа из 22 символов

13-й элемент:

01 – CONSTANT_Utf8 (константа)

0x000a – длина константы

SourceFile – строковая константа из 10 символов

14-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0008 – длина константы

App.java – строковая константа из 8 символов

15-й элемент:

0c – CONSTANT_NameAndType

0x0007 – ссылка на 7-й элемент пула <init> (конструктор без параметров)

0x0008 – ссылка на 8-й элемент пула ()V (описатель конструктора без параметров)

16-й элемент:

07 – CONSTANT_Class

0x0017 – название класса в 23 элементе пула (строка java/lang/System)

17-й элемент:

0c – CONSTANT_NameAndType

0x0018 – ссылка на 24-й элемент пула out

0x0019 – ссылка на 25-й элемент пула Ljava/io/PrintStream;

18-й элемент:

01 – CONSTANT_Utf8 (константа)

0x000c – длина константы

HelloWorld! – строковая константа из 12 символов

19-й элемент:

07 – CONSTANT_Class

0x001a – название класса в 25 элементе пула (строка из 21 символа Utf8:

java/io/PrintStream)

20-й элемент:

0с – CONSTANT_NameAndType

0x001b – ссылка на 27-й элемент пула `println`

0x001с – ссылка на 28-й элемент пула `(Ljava/lang/String;)V`

21-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0009 – длина константы

`hello/App` – строковая константа из 9 символов

22-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0010 – длина константы

`java/lang/Object` – строковая константа из 16 символов

23-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0010 – длина константы

`java/lang/System` – строковая константа из 16 символов

24-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0003 – длина константы

`out` – строковая константа из 3 символов

25-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0015 – длина константы

`Ljava/io/PrintStream;` – строковая константа из 21 символа

26-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0013 – длина константы

`java/io/PrintStream` – строковая константа из 19 символов

27-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0007 – длина константы

`println` – строковая константа из 7 символов

28-й элемент:

01 – CONSTANT_Utf8 (константа)

0x0015 – длина константы

`(Ljava/lang/String;)V` – строковая константа из 16 символов

Массив `constant_pool` закончился.

Далее идут флаги доступа. Это битовая маска для свойств модификаторов:

0021:

`access_flags` – ACC_PUBLIC, ACC_SUPER, так как класс публичный и при создании требуется вызвать метод суперкласса.

`this_class` – ссылка 0x0005 на имя класса в пуле констант `hello/app`

`super_class` – ссылка 0x0006 на родительский класс (`java/lang/Object`)

Далее следует описание количества интерфейсов, полей, методов и атрибутов
`interfaces: 0, fields: 0, methods: 2, attributes: 1`

В классе 2 метода: конструктор по умолчанию и метод `main`:

00 02 – `methods_count`

Затем в файле последовательно следует описание методов (их в классе 2)

	Method 1	Method 2
Constructor/main	00 01 – <code>access_flags</code> (public) 00 07 – <code>name_index</code> имя <init> 00 08 – <code>descriptor_index</code> ()V описатель конструктора, сигнатура 00 01 – <code>attributes_count</code>	00 09 – <code>access_flags</code> (super) 00 0b – <code>name_index</code> main 00 0c – <code>descriptor_index</code> ([Ljava/lang/String;)V 00 01 – <code>atributes_count</code>
Attribute 1	00 09 – <code>name_index</code> (Code) 00 00 00 1d – <code>attribute_length</code> 00 01 – <code>max_stack</code> размер стека нужный для выполнения операции 00 01 – <code>max_locals</code> размер локальных переменных 00 00 00 05 – <code>code_length</code> 5 байтов	00 09 – <code>name_index</code> (Code) 00 00 00 25 – <code>attribute_length</code> 00 02 – <code>max_stack</code> 00 01 – <code>max_locals</code> 00 00 00 09 – <code>code_length</code> 9 байтов
<code>code[code_length]</code> (атрибут)	2a – команда <code>aload_0</code> кладет <code>this</code> в стек b7 00 01 – запуск суперкласса <code>invokespecial #0001</code> (ссылка на класс в 6-м элементе пула – <code>java/lang/Object</code>) b1 – <code>return</code>	b2 00 02 – <code>getstatic #0002</code> (распутывая цепочку по индексам – <code>java.lang.System</code>) 12 03 – <code>ldc 3</code> в стек по #3, а там ссылка на строку <code>Hello World!</code> b6 00 04 – <code>invokevirtual #4</code> вызов экземпляра метода класса #4, в котором ссылки на имя <code>java/io/PrintStream</code> класса и метода в нем <code>println</code> с дескриптором ([Ljava/lang/String;)V b1 – <code>return</code>
Описание метода	00 00 – <code>exception_table_length</code> – число элементов в табл. перехватов исключений 00 01 – <code>attributes_count</code> в Code 00 0a – <code>attribute_name_index</code> #10 00 00 00 06 – <code>attribute_length</code> 00 01 – <code>line_number_table_length</code> 00 00 – <code>start_pc</code> 00 03 – <code>line_number</code>	00 00 – <code>exception_table_length</code> 00 01 – <code>attributes_count</code> 00 0a – <code>attribute_name_index</code> 00 00 00 0a – <code>attribute_length</code> 00 02 – <code>line_nuber_table_length</code> 00 00 – <code>start_pc</code> 00 06 – <code>line_number</code> 00 08 – <code>start_pc</code> 00 07 – <code>line_number</code>

В конце файла следует описание атрибутов класса:

```
00 01          - attributes_count
00 0d          - name_index (SourceFile)
00 00 00 02    - attributes_length
00 0e          - sourcefile_index(App.java)
```