public class JvmComprehension {

public static void main(String[] args) {

int i = 1; // 1

Object o = new Object(); // 2

Integer ii = 2; // 3

printAll(o, i, ii); // 4

System.out.println("finished"); // 7

}

private static void printAll(Object o, int i, Integer ii) {

Integer uselessVar = 700; // 5

System.out.println(o.toString() + i + ii); // 6

}

}

Загрузка класса JvmComprehension в Application ClassLoader, далеее проверка валидности, связывание – Linkiding и инициализация static полей Initialization, потом загрузка в область памяти Metaspace (загружаются данные о классе и константы)

Runtime Data Area. Создается стековая память для использования методом main(). Создается фрейм main() в Stack Memory

1. Выделяется область памяти внутри фрейма main() в Stack Memory под int I и присваивается этой переменной значение – 1
2. В heap (куча) выделяется область памяти под объект Object, в стеке внутри фрейма main() выделяется память под переменную – о, в которую записывается адрес в куче объекта Object
3. В области памяти heap (куча) выделяется место под объект Integer, создается ссылка ii в памяти стека фрейма main() на переменную Integer, значение которой = 2 помещается в heap.
4. создается фрейм printAll() в Stack Memory, в который передается ссылки на o, ii и переменная int i
5. В heap выделяется блок памяти под Integer, а в памяти стека фрейма printAll() создается ссылка на него uselessVar. Значение 700 помещается в heap
6. В стеке создается фрейм для метода println() и в нем переменная int i, ссылка ii объекта Integer, созданный в heap п.3, ссылка на строку, которую вернет метод toString(). В heap под эту строку выделяется память String

В стеке создается фрейм toString() и в нем ссылка о на объект Object, созданный в heap п.п. 2. Метод toString() возвращает строку, фрейм закрывается. Метод println() выводит на печать, фрейм закрывается. Метод printAll() отработал, фрейм закрывается.

1. Рисунок 2. В стеке создается фрейм памяти для метода println() и в нем ссылка на строку "finished", сама строка попадает в String. Строка выводится на печать, закрывается фрейм метода println(). Закрывается фрейм метода main().

Сборщик мусора сначала удалит неиспользуемую переменную uselessVar, затем строку из String после выполнения метода println() п.6,затем Object и строку из String после выполнения метода println() п.7