Chapter 1: Text Blocks, Locales, Numbers, and Math

```
-- BEFORE TEXT BLOCK --
UPDATE "public"."office"
SET ("address_first", "address_second", "phone") =
  (SELECT "public"."employee"."first_name",
          "public"."employee"."last_name", ?
   FROM "public"."employee"
   WHERE "public"."employee"."job_title" = ?
-- AFTER TEXT BLOCK --
```

A

```
-- BEFORE TEXT BLOCK --
UPDATE "public"."office"
SET ("address_first", "address_second", "phone") =
  (SELECT "public"."employee"."first_name",
          "public"."employee"."last_name", ?
   FROM "public"."employee"
   WHERE "public"."employee"."job_title" = ?

-- AFTER TEXT BLOCK --
```

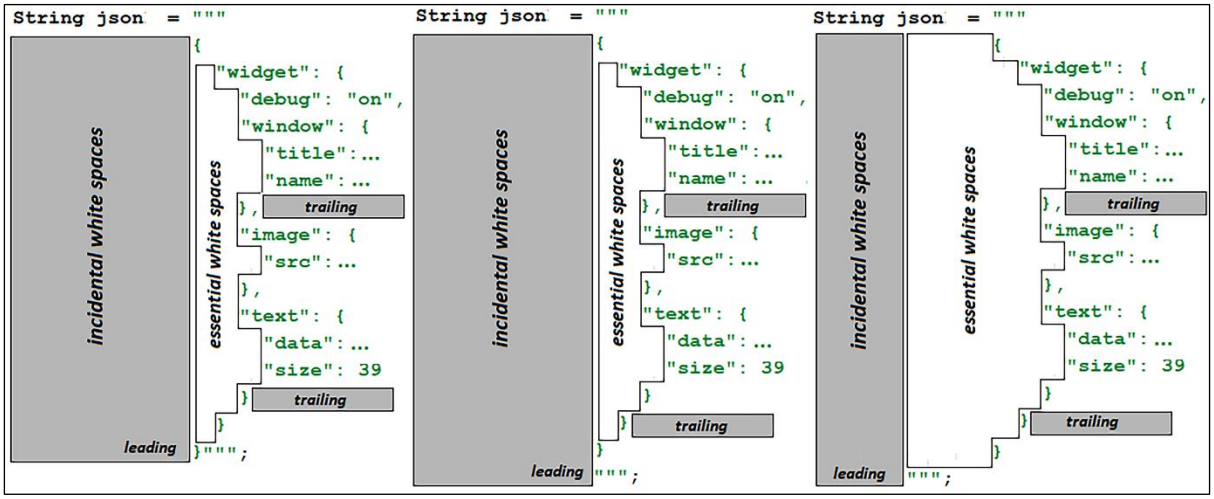
B

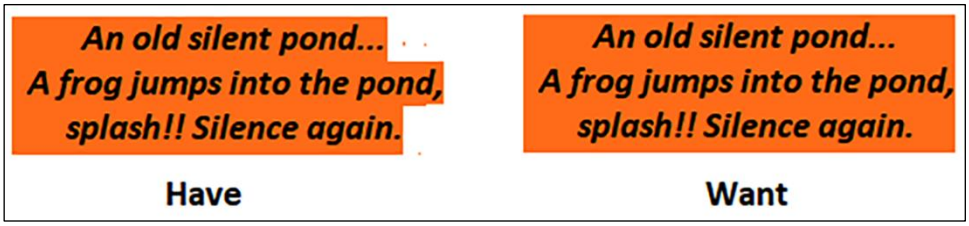
```
-- BEFORE TEXT BLOCK --
UPDATE "public"."office"
SET ("address_first", "address_second", "phone") =
  (SELECT "public"."employee"."first_name",
          "public"."employee"."last_name", ?
   FROM "public"."employee"
   WHERE "public"."employee"."job_title" = ?

-- AFTER TEXT BLOCK --
```

A

```
-- BEFORE TEXT BLOCK --
UPDATE "public"."office"
SET ("address_first", "address_second", "phone") =
  (SELECT "public"."employee"."first_name",
          "public"."employee"."last_name", ?
   FROM "public"."employee"
   WHERE "public"."employee"."job_title" = ?

-- AFTER TEXT BLOCK --
```

B





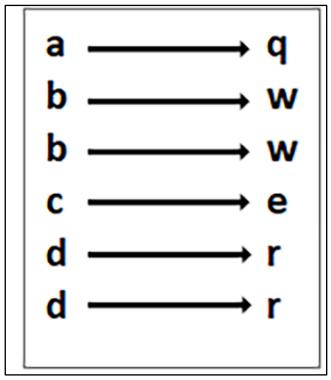
```
JASON MILLER (\"BIGBOY\")\n
\tMOUNT INC\n
\t104 SEAL AVE\n
\tMIAMI FL 55334 1200\n
\tUSA
```

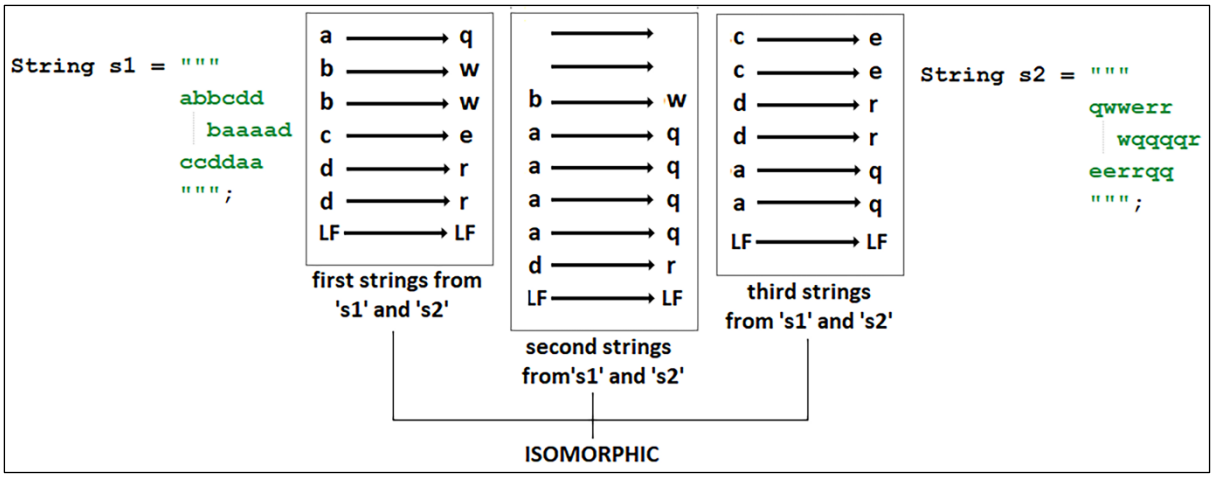
```
JASON MILLER ("BIGBOY")

        MOUNT INC

        104 SEAL AVE

        MIAMI FL 55334 1200

        USA
```

```
<user>
   <firstName>Jo</firstName>
   <lastName>Kym</lastName>
</user>
```

```
<user>
   <firstName>
    Jo
   </firstName>
   <lastName>
    Kym
   </lastName>
</user>
```

```
Benchmark                Mode  Cnt     Score      Error  Units
Main.concatenation       avgt    5     0.670 ±    0.004  ns/op
Main.messageFormat       avgt    5  3199.252 ±  116.859  ns/op
Main.stringBuilder       avgt    5   318.641 ±    9.198  ns/op
Main.stringFormat        avgt    5  1416.266 ±   18.024  ns/op
Main.stringFormatted     avgt    5  1409.030 ±   42.547  ns/op
```






```java
public static String concatViaPlus(
        String str1, String str2,
        String str3, String str4) {

    return str1 + str2 + str3 + str4;
}
```

```java
public static String concatViaStringBuilder(
        String str1, String str2,
        String str3, String str4) {

    StringBuilder sb = new StringBuilder();

    sb.append(str1)
            .append(str2)
            .append(str3)
            .append(str4);

    return sb.toString();
}
```

```
JDK 8 bytecode of Strings#concatViaPlus()
------------------------------------------------------------
Code(maxStack = 2, maxLocals = 4, code_length = 27)
0:    new               <java.lang.StringBuilder> (14)
3:    dup
4:    invokespecial     java.lang.StringBuilder.<init> ()V (16)
7:    aload_0
8:    invokevirtual     java.lang.StringBuilder.append (Ljava/lang/String;)Ljava/lang/StringBuilder; (17)
11:   aload_1
12:   invokevirtual     java.lang.StringBuilder.append (Ljava/lang/String;)Ljava/lang/StringBuilder; (17)
15:   aload_2
16:   invokevirtual     java.lang.StringBuilder.append (Ljava/lang/String;)Ljava/lang/StringBuilder; (17)
19:   aload_3
20:   invokevirtual     java.lang.StringBuilder.append (Ljava/lang/String;)Ljava/lang/StringBuilder; (17)
23:   invokevirtual     java.lang.StringBuilder.toString ()Ljava/lang/String; (21)
26:   areturn
```

Benchmark	Mode	Cnt	Score	Error	Units
Main.concatViaPlus	avgt	90	37.370	± 0.215	ns/op
Main.concatViaStringBuilder	avgt	90	78.841	± 1.000	ns/op

```
JDK 11 bytecode of Strings#concatViaPlus()
------------------------------------------------------------
Code(maxStack = 4, maxLocals = 4, code_length = 10)
0:    aload_0
1:    aload_1
2:    aload_2
3:    aload_3
4:    invokedynamic     0:makeConcatWithConstants (Ljava/lang/String;Ljava/lang/String;
                                                   Ljava/lang/String;Ljava/lang/String;)
                                                   Ljava/lang/String; (14)     00

9:    areturn
```

```
JDK 11 bytecode of Strings#concatViaStringBuilder()
------------------------------------------------------------
Code(maxStack = 2, maxLocals = 5, code_length = 34)
0:    new               <java.lang.StringBuilder> (18)
3:    dup
4:    invokespecial     java.lang.StringBuilder.<init> ()V (20)
7:    astore            %4
9:    aload             %4
11:   aload_0
12:   invokevirtual     java.lang.StringBuilder.append (Ljava/lang/String;)Ljava/lang/StringBuilder; (21)
15:   aload_1
16:   invokevirtual     java.lang.StringBuilder.append (Ljava/lang/String;)Ljava/lang/StringBuilder; (21)
19:   aload_2
20:   invokevirtual     java.lang.StringBuilder.append (Ljava/lang/String;)Ljava/lang/StringBuilder; (21)
23:   aload_3
24:   invokevirtual     java.lang.StringBuilder.append (Ljava/lang/String;)Ljava/lang/StringBuilder; (21)
27:   pop
28:   aload             %4
30:   invokevirtual     java.lang.StringBuilder.toString ()Ljava/lang/String; (25)
33:   areturn
```

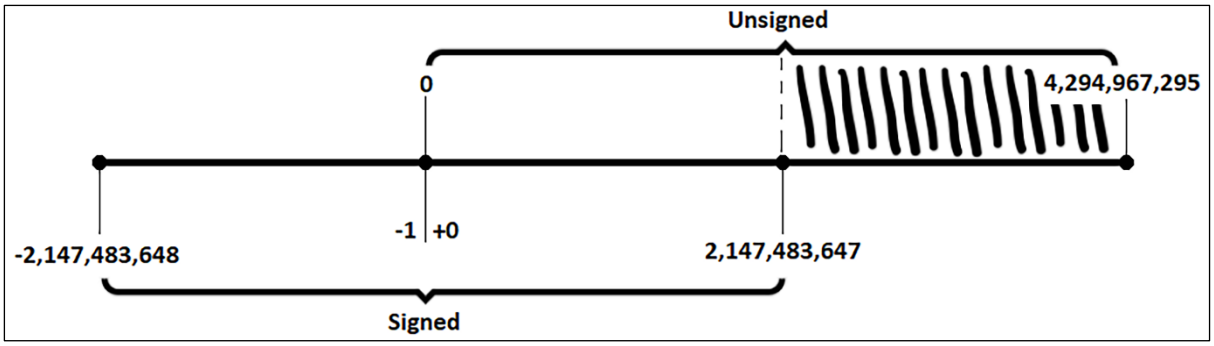
Benchmark	Mode	Cnt	Score	Error	Units
Main.concatViaPlus	avgt	90	47.825	± 0.177	ns/op
Main.concatViaStringBuilder	avgt	90	76.861	± 0.451	ns/op

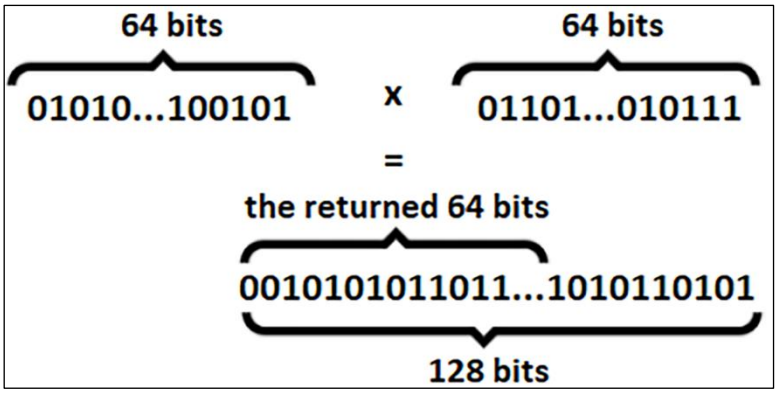
Benchmark	Mode	Cnt	Score	Error	Units
Main.intToStringV1	avgt	20	19.499	± 0.428	ns/op
Main.intToStringV2	avgt	20	20.145	± 0.243	ns/op
Main.intToStringV3	avgt	20	19.207	± 0.148	ns/op
Main.intToStringV4	avgt	20	171.027	± 0.641	ns/op
Main.integerToStringV1	avgt	20	27.255	± 0.207	ns/op
Main.integerToStringV2	avgt	20	29.618	± 0.056	ns/op
Main.integerToStringV3	avgt	20	27.231	± 0.064	ns/op
Main.integerToStringV4	avgt	20	166.432	± 0.337	ns/op

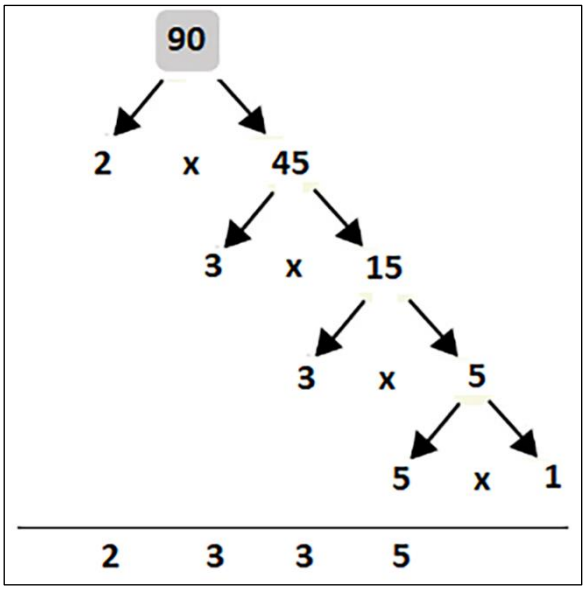
	ofLocalizedDate	ofLocalizedTime	ofLocalizedDateTime
LocalTime		MEDIUM, SHORT	
LocalDate	FULL, LONG, MEDIUM, SHORT		
LocalDateTime	FULL, LONG, MEDIUM, SHORT	MEDIUM, SHORT	MEDIUM, SHORT
ZonedDateTime	FULL, LONG, MEDIUM, SHORT	FULL, LONG, MEDIUM, SHORT	FULL, LONG, MEDIUM, SHORT
OffsetDateTime	FULL, LONG, MEDIUM, SHORT	MEDIUM, SHORT	MEDIUM, SHORT

```
"G{0,5}" +        // Era
"y*" +            // Year
"Q{0,5}" +        // Quarter
"M{0,5}" +        // Month
"w*" +            // Week of Week Based Year
"E{0,5}" +        // Day of Week
"d{0,2}" +        // Day of Month
"B{0,5}" +        // Period/AmPm of Day
"[hHjC]{0,2}" +   // Hour of Day/AmPm (refer to LDML for 'j' and 'C')
"m{0,2}" +        // Minute of Hour
"s{0,2}" +        // Second of Minute
"[vz]{0,4}"       // Zone
```

```
Benchmark                    (v)  Mode  Cnt   Score   Error  Units
Main.isDoubleIntegerV1     23.11  avgt   10   1.697 ± 0.022  ns/op
Main.isDoubleIntegerV1        23  avgt   10   1.671 ± 0.014  ns/op
Main.isDoubleIntegerV2     23.11  avgt   10  23.985 ± 0.406  ns/op
Main.isDoubleIntegerV2        23  avgt   10  18.384 ± 0.196  ns/op
Main.isDoubleIntegerV3     23.11  avgt   10   1.338 ± 0.009  ns/op
Main.isDoubleIntegerV3        23  avgt   10   1.667 ± 0.021  ns/op
Main.isDoubleIntegerV4     23.11  avgt   10   1.169 ± 0.024  ns/op
Main.isDoubleIntegerV4        23  avgt   10   1.702 ± 0.068  ns/op
Main.isDoubleIntegerV5     23.11  avgt   10   1.329 ± 0.007  ns/op
Main.isDoubleIntegerV5        23  avgt   10   1.665 ± 0.016  ns/op
Main.isDoubleIntegerV6     23.11  avgt   10   2.394 ± 0.023  ns/op
Main.isDoubleIntegerV6        23  avgt   10   2.400 ± 0.024  ns/op
```





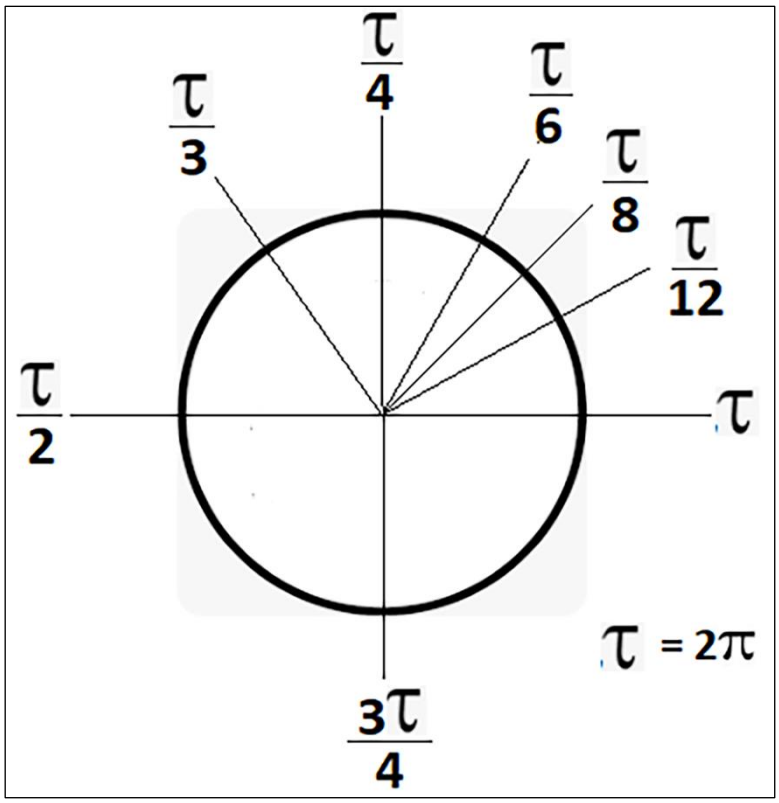


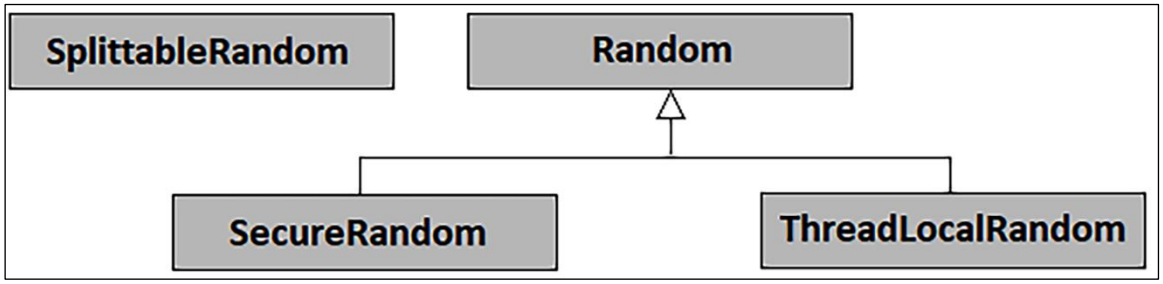
$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{v}{x_n}\right) = F(x_n)$$

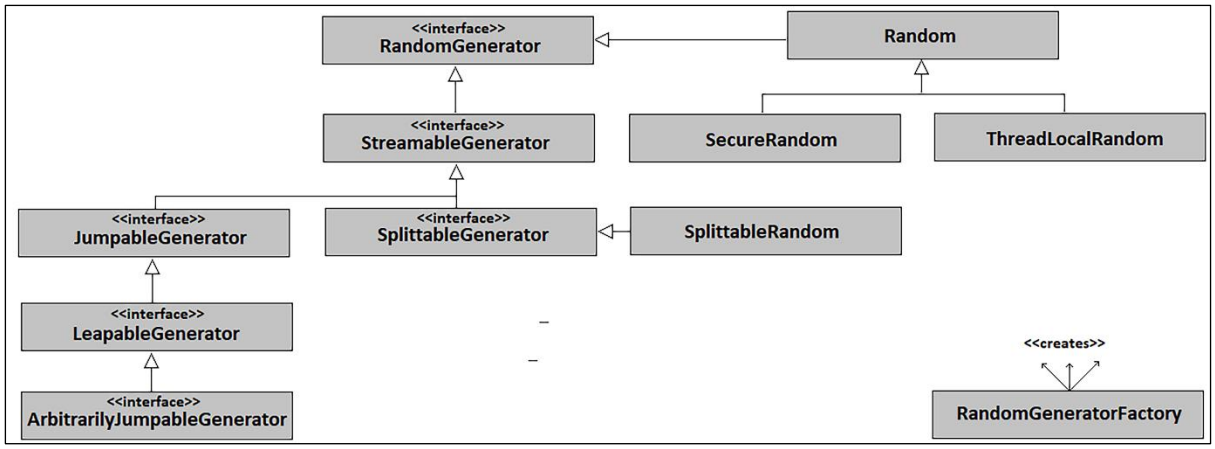
$$(a+b)^2 = a^2 + b^2 + 2ab$$

$$\Downarrow$$

$$ab = ((a+b)^2 - a^2 - b^2)/2$$

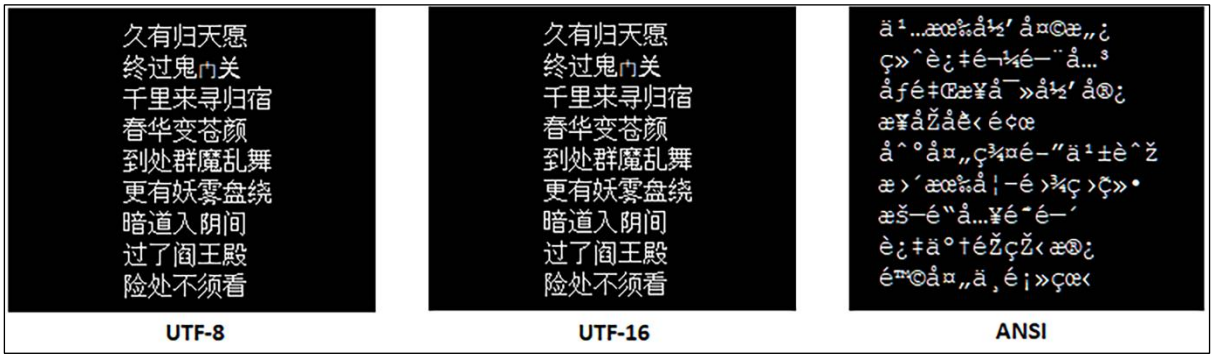






Name	Group	Ar. Jumpable	Deprecated	Hardware	Jumpable	Leapable	Splittable	Statistical	Stochastic	Streamable
L32X64MixRandom	LXM	false	false	false	false	false	true	true	false	true
L128X128MixRandom	LXM	false	false	false	false	false	true	true	false	true
L64X128MixRandom	LXM	false	false	false	false	false	true	true	false	true
L128X1024MixRandom	LXM	false	false	false	false	false	true	true	false	true
L64X128StarStarRandom	LXM	false	false	false	false	false	true	true	false	true
L64X256MixRandom	LXM	false	false	false	false	false	true	true	false	true
L128X256MixRandom	LXM	false	false	false	false	false	true	true	false	true
L64X1024MixRandom	LXM	false	false	false	false	false	true	true	false	true
SecureRandom	Legacy	false	false	false	false	false	false	false	true	false
Random	Legacy	false	false	false	false	false	false	true	false	false
SplittableRandom	Legacy	false	false	false	false	false	true	true	false	true
Xoroshiro128PlusPlus	Xoroshiro	false	false	false	true	true	false	true	false	true
Xoshiro256PlusPlus	Xoshiro	false	false	false	true	true	false	true	false	true

Chapter 2: Objects, Immutability, Switch Expressions, and Pattern Matching



$0\,0\,0\,0\,0\,0\,0 = 0$
$2^6\,2^5\,2^4\,2^3\,2^2\,2^1\,2^0$

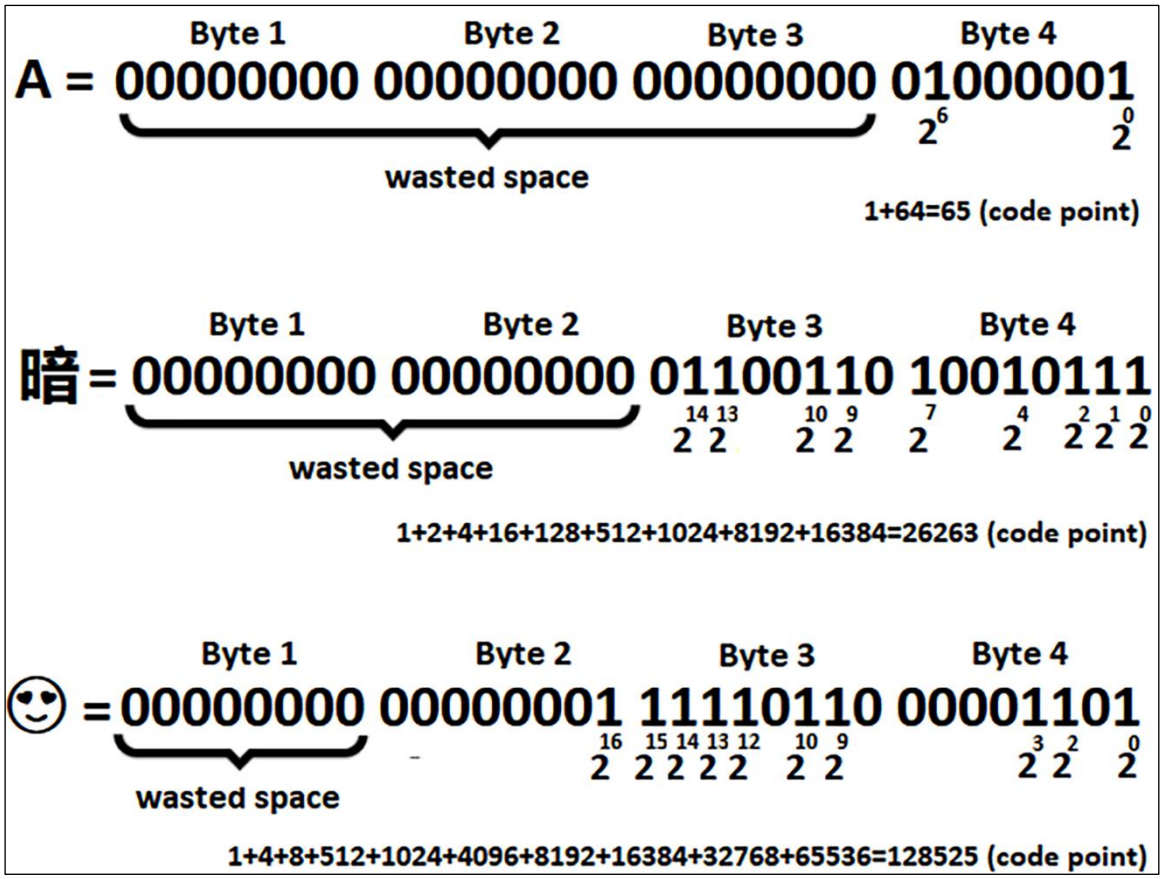
$1\,1\,1\,1\,1\,1\,1 = 1 + 2 + 4 + 8 + 16 + 32 + 64 = 127$
$2^6\,2^5\,2^4\,2^3\,2^2\,2^1\,2^0$

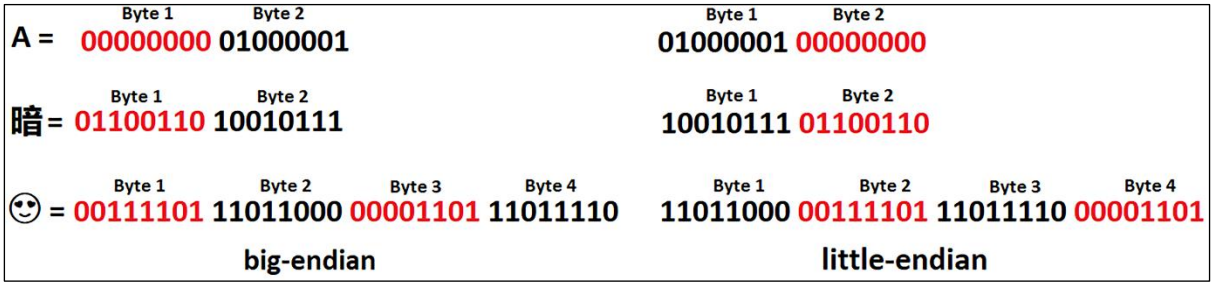
A = 65 = 1000001
B = 66 = 1000010
C = 67 = 1000011
...
a = 97 = 1100001
b = 98 = 1100010
c = 99 = 1100011
...

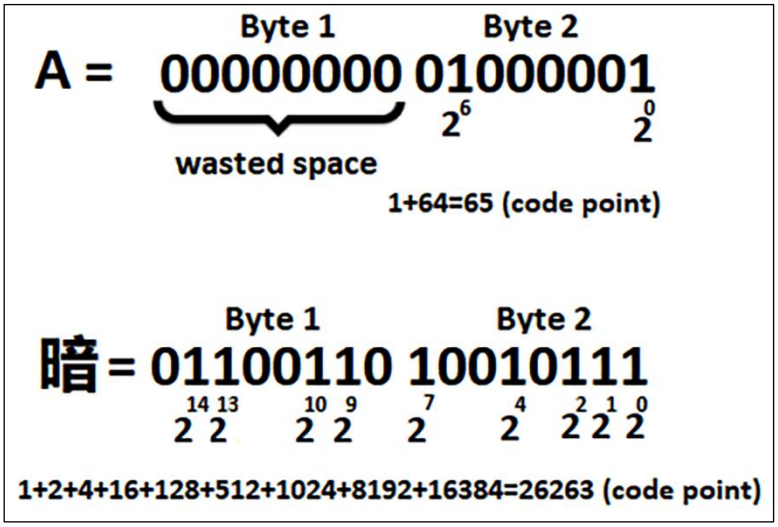
$0\,0\,0\,0\,0\,0\,0\,0 = 0$
$2^7\,2^6\,2^5\,2^4\,2^3\,2^2\,2^1\,2^0$

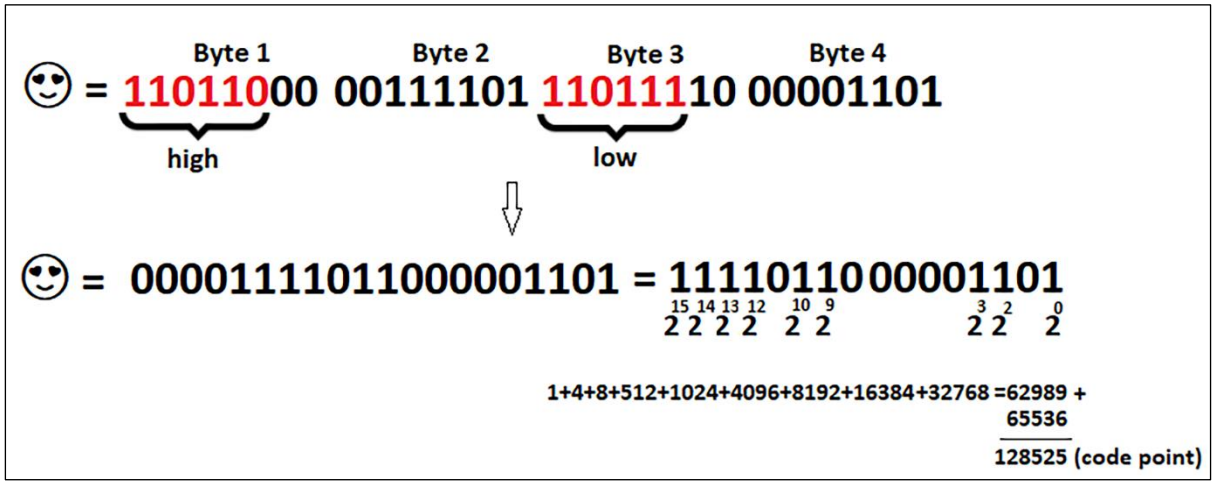
$0\,1\,1\,1\,1\,1\,1\,1 = 1 + 2 + 4 + 8 + 16 + 32 + 64 = 127$
$2^7\,2^6\,2^5\,2^4\,2^3\,2^2\,2^1\,2^0$

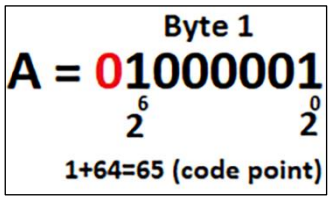
A = 65 = 01000001
B = 66 = 01000010
C = 67 = 01000011
...
a = 97 = 01100001
b = 98 = 01100010
c = 99 = 01100011
...

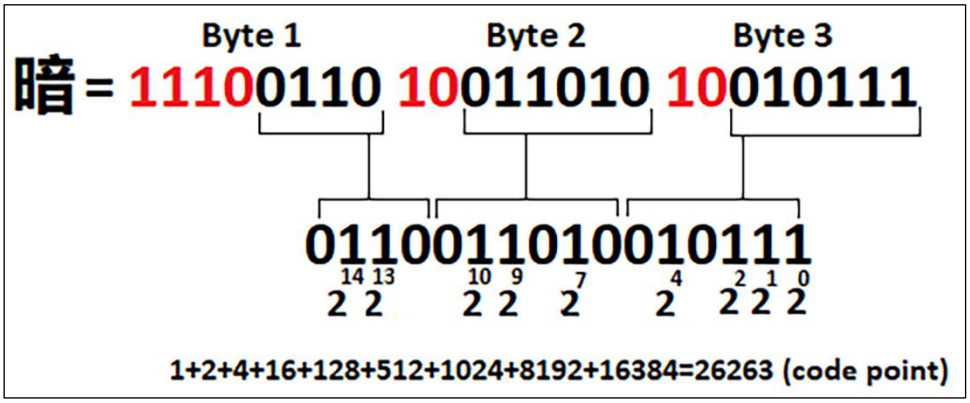

Byte 1 Byte 2 Byte 3 Byte 4
A = 00000000 00000000 00000000 01000001
wasted space
1+64=65 (code point)
Byte 1 Byte 2 Byte 3 Byte 4
暗 = 00000000 00000000 01100110 10010111
wasted space
1+2+4+16+128+512+1024+8192+16384=26263 (code point)
Byte 1 Byte 2 Byte 3 Byte 4
= 00000000 00000001 11110110 00001101
wasted space
1+4+8+512+1024+4096+8192+16384+32768+65536=128525 (code point)

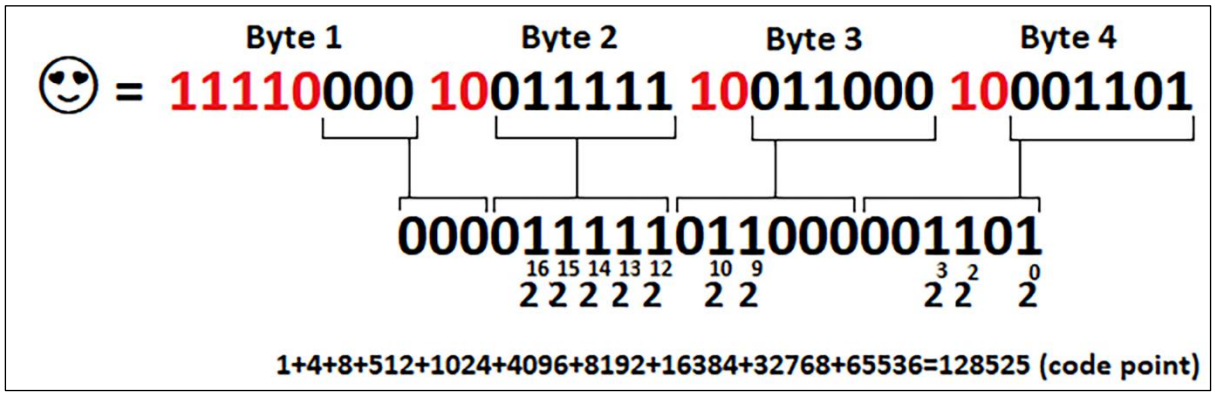

A = Byte 1 Byte 2
00000000 01000001
01000001 00000000
暗 = Byte 1 Byte 2
01100110 10010111
10010111 01100110
= Byte 1 Byte 2 Byte 3 Byte 4
00111101 11011000 00001101 11011110
11011000 00111101 11011110 00001101
big-endian
little-endian


Byte 1 Byte 2
A = 00000000 01000001
wasted space
1+64=65 (code point)
Byte 1 Byte 2
暗 = 01100110 10010111
1+2+4+16+128+512+1024+8192+16384=26263 (code point)


Byte 1 Byte 2 Byte 3 Byte 4
= 11011000 00111101 11011110 00001101
high
low
= 0000111101100001101 = 1111011000001101
15 14 13 12 10 9 3 2 0
2 2 2 2 2 2 2 2 2
1+4+8+512+1024+4096+8192+16384+32768 =62989 +
65536
128525 (code point)


Byte 1
A = 01000001
6 0
2 2
1+64=65 (code point)


Byte 1 Byte 2 Byte 3
暗 = 11100110 10011010 10010111
0110011010010111
14 13 10 9 7 4 2 1 0
2 2 2 2 2 2 2 2 2
1+2+4+16+128+512+1024+8192+16384=26263 (code point)


Byte 1 Byte 2 Byte 3 Byte 4
= 11110000 10011111 10011000 10001101
00001111011000001101
16 15 14 13 12 10 9 3 2 0
2 2 2 2 2 2 2 2 2 2
1+4+8+512+1024+4096+8192+16384+32768+65536=128525 (code point)

Number of bytes	First code point	Last code point	Byte 1	Byte 2	Byte 3	Byte 4
1	U+0000	U+007F	0xxxxxxx			
2	U+0080	U+07FF	110xxxxx	10xxxxxx		
3	U+0800	U+FFFF	1110xxxx	10xxxxxx	10xxxxxx	
4	U+10000	U+1FFFFF	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx

```
int a1 = "A".charAt(0);          // 65
int a2 = "A".codePointAt(0);     // 65

int b1 = "暗".charAt(0);         // 26263
int b2 = "暗".codePointAt(0);    // 26263

int c1 = "😍".charAt(0);         // 55357  - wrong
int c2 = "😍".codePointAt(0);    // 128525 - correct
```

A telemeter with laser ranging from 0 to 60 ft including calculation of surfaces and volumes with high-precision

```
Telemeter.Calibrate.at(0.00001);
Telemeter telemeter = new Telemeter(0.15, 2, "IP54");
```

```
* {@snippet :
*      Telemeter.Calibrate.at(0.00001);
*      Telemeter telemeter = new Telemeter(
*          0.15,
*          2,
*          "IP54"
*      );
* }
```

```
Telemeter.Calibrate.at(0.00001);
Telemeter telemeter = new Telemeter(
   0.15,
   2,
   "IP54"
);
```

```
* {@snippet :
*      Telemeter.Calibrate.at(0.00001);
*      Telemeter telemeter = new Telemeter(
*          0.15,
*          2,
*          "IP54"
*      );
*      }
```

```
Telemeter.Calibrate.at(0.00001);
Telemeter telemeter = new Telemeter(
   0.15,
   2,
   "IP54"
);
```

```
* {@snippet lang="java" :
*      Telemeter.Calibrate.at(0.00001);  // @highlight
*      Telemeter telemeter = new Telemeter(0.15, 2, "IP54");
* }
```

```
Telemeter.Calibrate.at(0.00001);
Telemeter telemeter = new Telemeter(0.15, 2, "IP54");
```

```
* {@snippet lang="java" :
*     Telemeter.Calibrate.at(0.00001);
*     Telemeter telemeter = new Telemeter( // @highlight region
*        0.15,
*        2,
*        "IP54"                           // @end
*     );
* }

  {@snippet lang="java" :
      Telemeter.Calibrate.at(0.00001);
      Telemeter telemeter = new Telemeter( // @highlight region=R1
         0.15,
         2,
         "IP54"                           // @end region=R1
      );
  }
```

```
Telemeter.Calibrate.at(0.00001);
Telemeter telemeter = new Telemeter(
   0.15,
   2,
   "IP54"
);
```

```
* {@snippet :
*  Telemeter.Calibrate.at(0.00001, "HIGH"); // @highlight substring="Calibrate" type=highlighted
* }
```

```
Telemeter.Calibrate.at(0.00001, "HIGH");
```

```
* Telemeter.Calibrate.at(0.00001); // @link substring="Calibrate" target="Calibrate#at(double, String)"
```

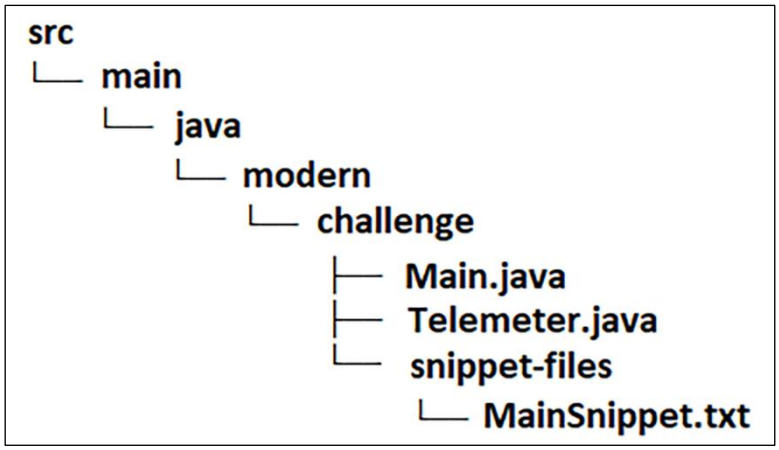
```
Telemeter.Calibrate.at(0.00001);
```

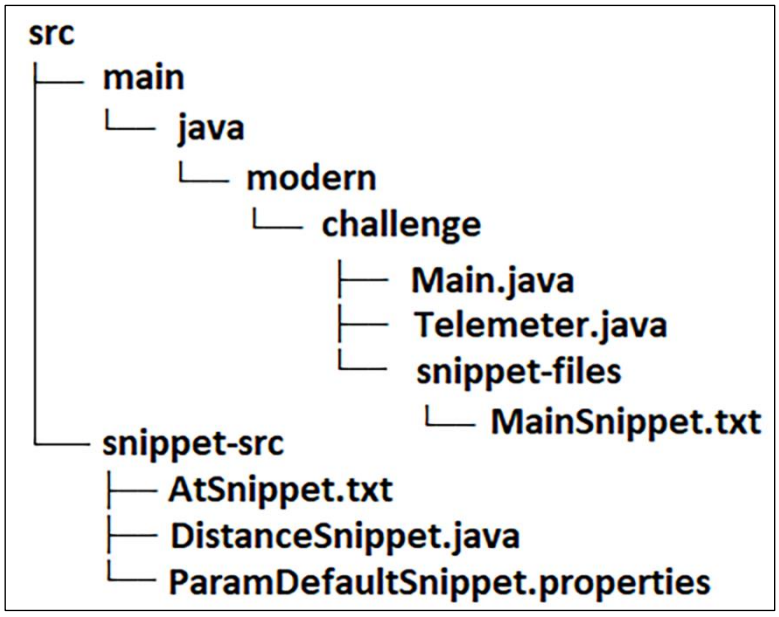
```
*  Telemeter.Calibrate.at(0.00001, "HIGH"); // @replace substring='0.00001' replacement="eps"
```

```
Telemeter.Calibrate.at(eps, "HIGH");
```

```
* Telemeter telemeter = new Telemeter( // @replace region regex="\d(?=\,)" replacement="..."
*     0.18,
*     7,
*     "IP54"                                // @end
* );
```

```
Telemeter telemeter = new Telemeter(
   0.1...,
   ...,
   "IP54"
);
```





```
 ...
 * <p><b>Defaults:</b></p>
 * {@snippet file = ParamDefaultSnippet.properties region=dist}
 */
public int distance(Point sc, Point ec, boolean interpolation) {
    ...
}
```

Defaults:

```
sc=[0,0]
ec=[0,0]
interpolation=false
```

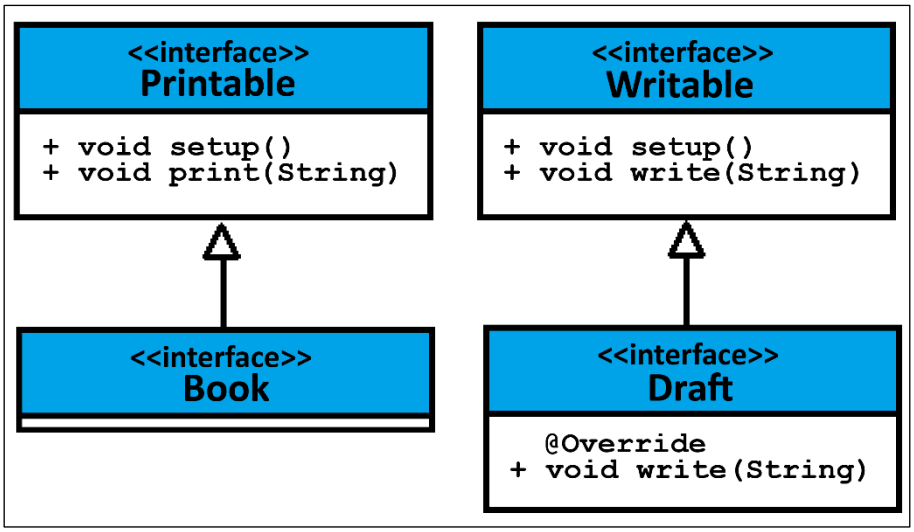
```
 ...
 * <p><b>Defaults:</b></p>
 * {@snippet file = ParamDefaultSnippet.properties region=at}
 *
 * PERFORM THE CALIBRATION BEFORE CREATING AN INSTANCE OF THIS CLASS
 */
public static boolean at(double eps, String type) {
    ...
}
```

Defaults:

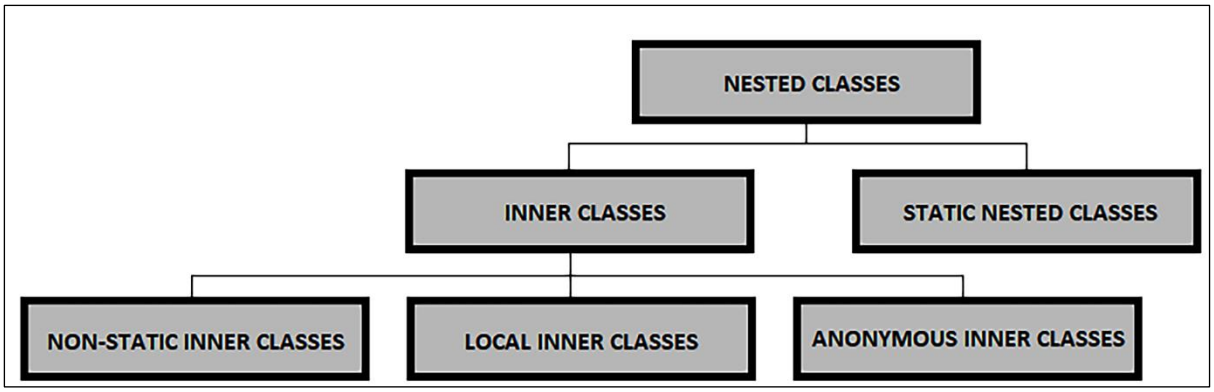
```
eps=0.1
type=null
```

PERFORM THE CALIBRATION BEFORE CREATING AN INSTANCE OF THIS CLASS

```
// This is an example used in the documentation
// @start region=only-code
                                    // @highlight region substring="Calibrate" type=highlighted
Telemeter.Calibrate.at(0.00001, "HIGH");  // @replace substring='0.00001' replacement="eps" @replace substring='"HIGH"' replacement='"..."'
                                    // @end
// @end region=only-code
```



Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Binary	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F




```
class to extend   constructor arguments
new ParentClass (args) {
    ...
    methods
};
```

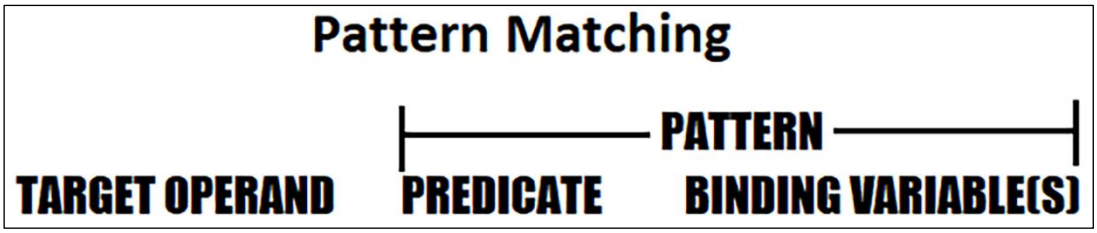
```
interface to implement
new InterfaceName() {
    ...
    methods
};
```

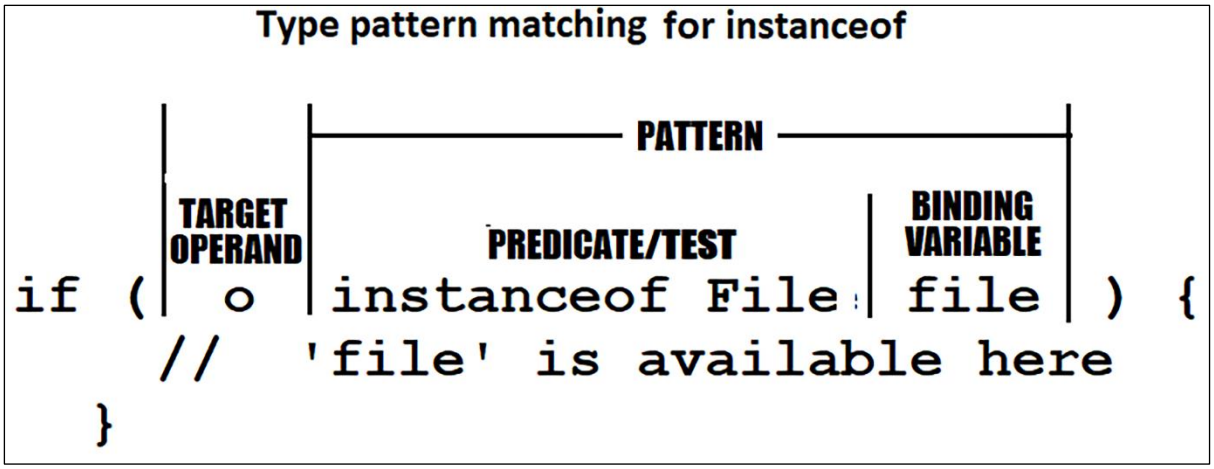
```
public class Main {
```

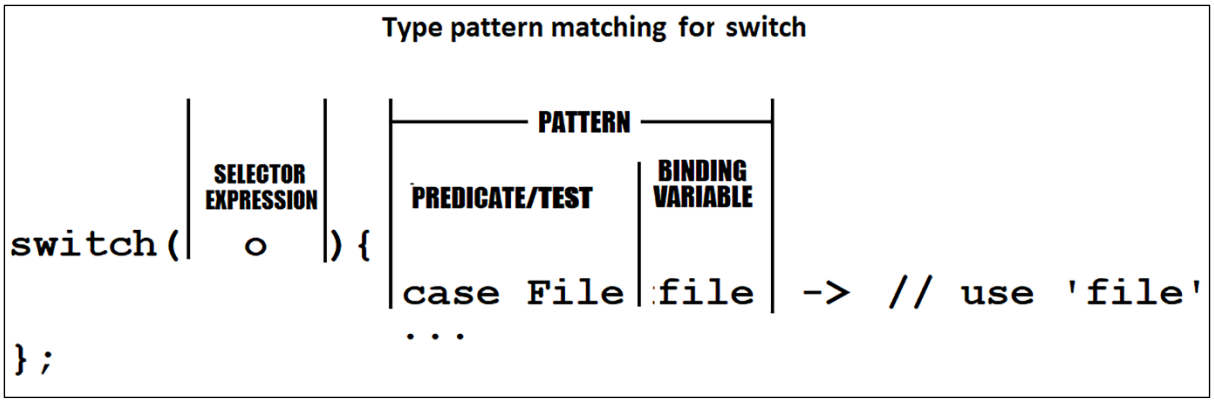
[missing-explicit-ctor] class Main in exported package modern.challenge declares no explicit constructors, thereby exposing a default constructor to clients of module P48_XlintDefaultConstructor

---

(Alt-Enter shows hints)

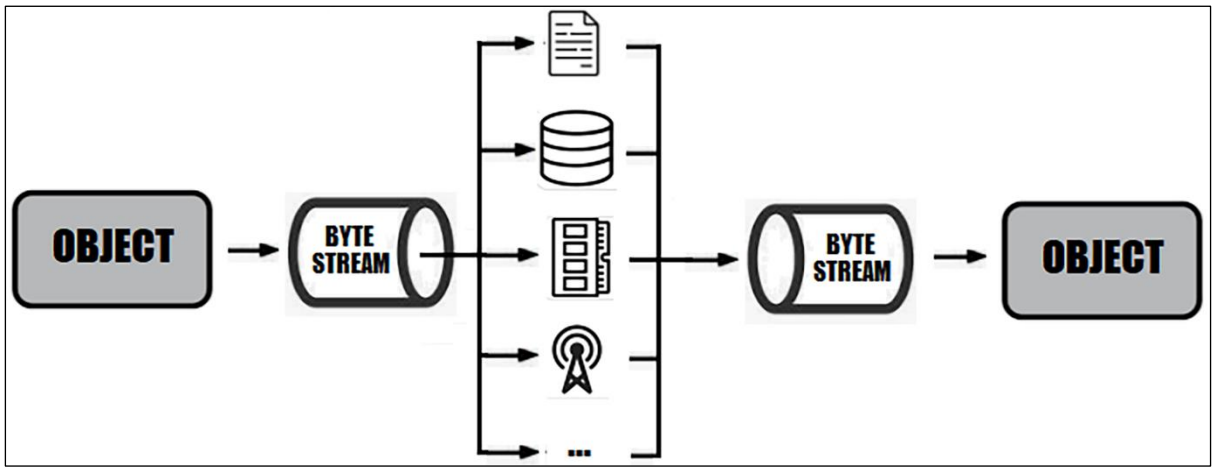

Pattern Matching
PATTERN
TARGET OPERAND
PREDICATE
BINDING VARIABLE(S)


Type pattern matching for instanceof
PATTERN
TARGET OPERAND
PREDICATE/TEST
BINDING VARIABLE
if (o instanceof File file) {
// 'file' is available here
}

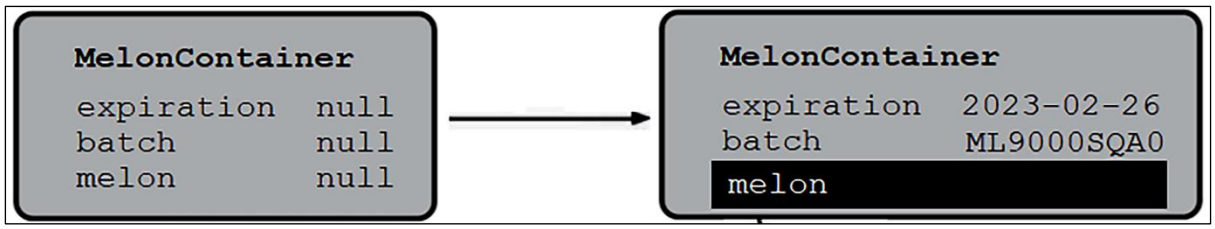

Type pattern matching for switch
PATTERN
SELECTOR EXPRESSION
PREDICATE/TEST
BINDING VARIABLE
switch(o) {
case File file -> // use 'file'
...
};

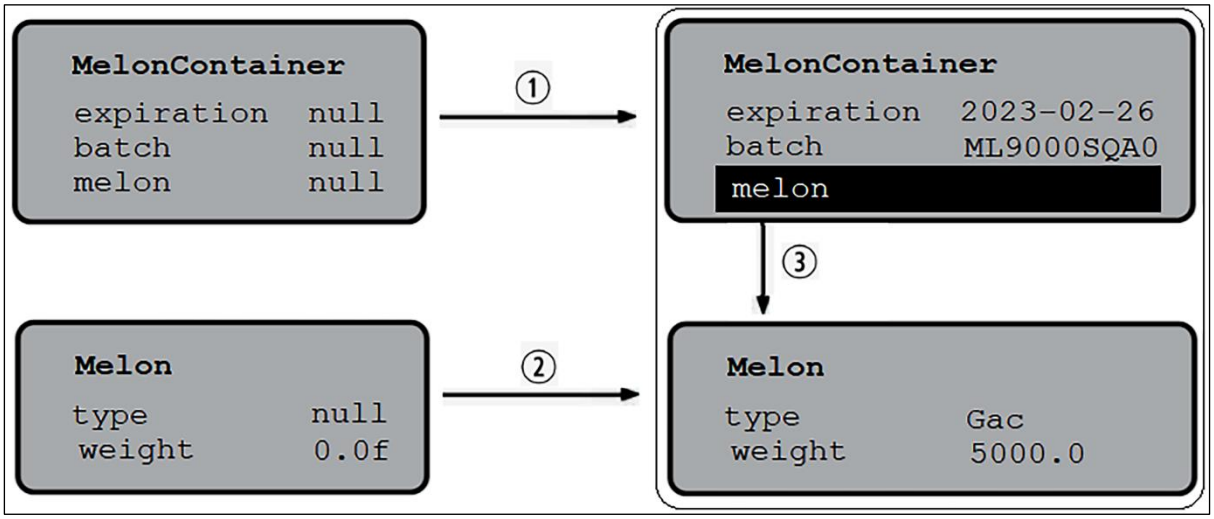
Chapter 4: Records and Record Patterns

```
public final class modern.challenge.MelonRecord extends java.lang.Record {
  public modern.challenge.MelonRecord(java.lang.String, float);
  public final java.lang.String toString();
  public final int hashCode();
  public final boolean equals(java.lang.Object);
  public java.lang.String type();
  public float weight();
}
```




```
OBJECT_A
   class_name:MelonContainer
   class_data [
    field:expiration, type:LocalDate, value:2023-02-26
    field:batch, type:String, value:ML9000SQA0
    field: melon, type:Melon, value:
       OBJECT_B
          class_name:Melon
          class_data [
           field:type, type:String, value:Gac
           field:weight, type:float, value:5000.0
          ]
    ]
```





Original stream

```
00000080  6D 6F 64 65 72 6E 2F 63 68 61 6C 6C 65 6E 67 65  modern/challenge
00000090  2F 4D 65 6C 6F 6E 3B 78 70 74 00 0A 4D 4C 39 30  /Melon;xpt..ML90
000000A0  30 30 53 51 41 30 73 72 00 0D 6A 61 76 61 2E 74  00SQA@sr..java.t
```

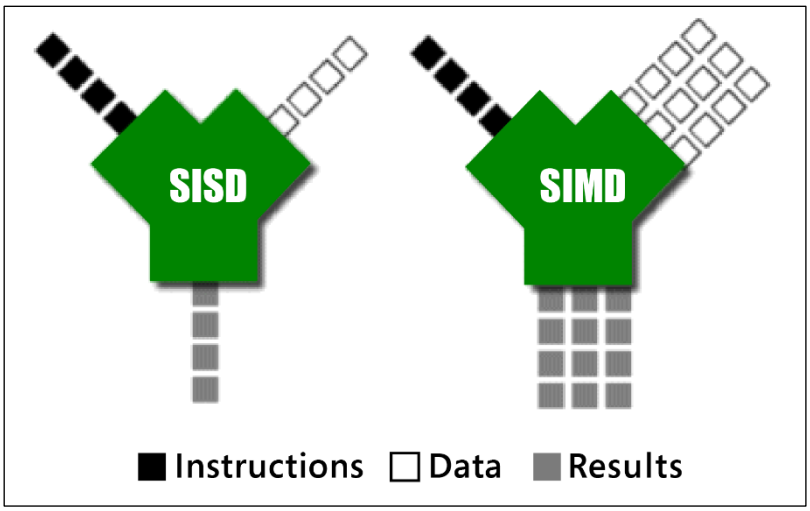
Malicious stream

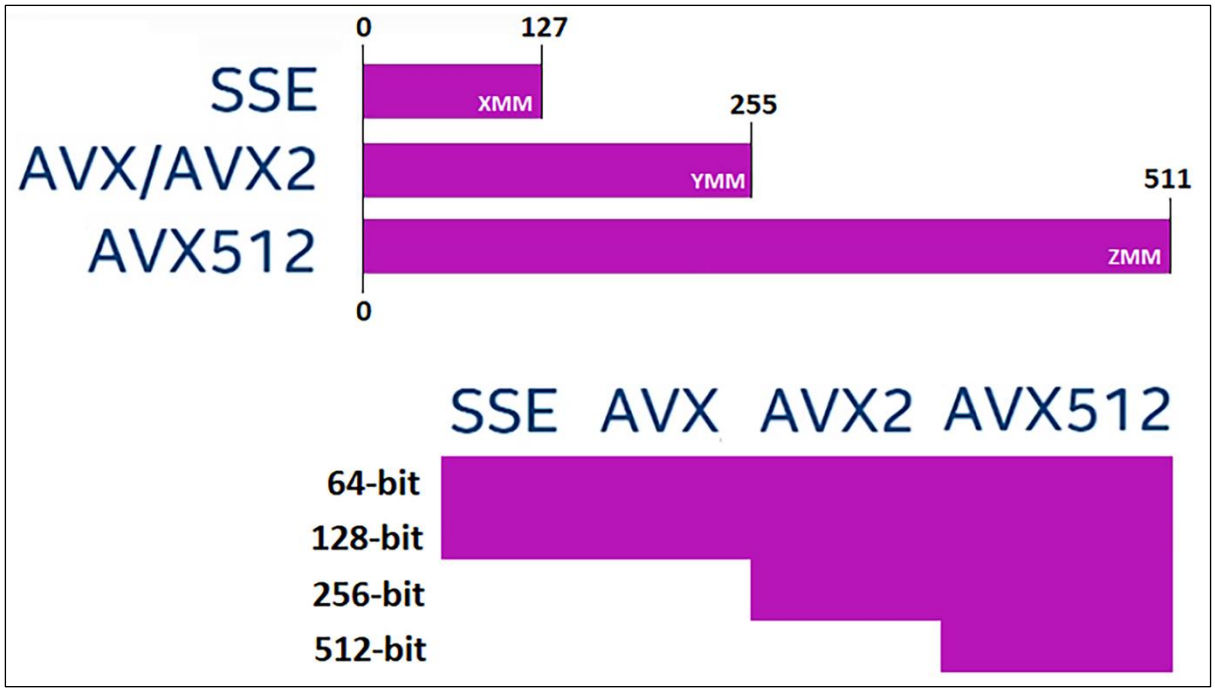
```
00000080  6D 6F 64 65 72 6E 2F 63 68 61 6C 6C 65 6E 67 65  modern/challenge
00000090  2F 4D 65 6C 6E 6E 3B 78 70 74 00 0A 30 30 30 30  /Melon;xpt..0000
000000A0  30 30 30 30 73 72 00 0D 6A 61 76 61 2E 74        000000sr..java.t
```

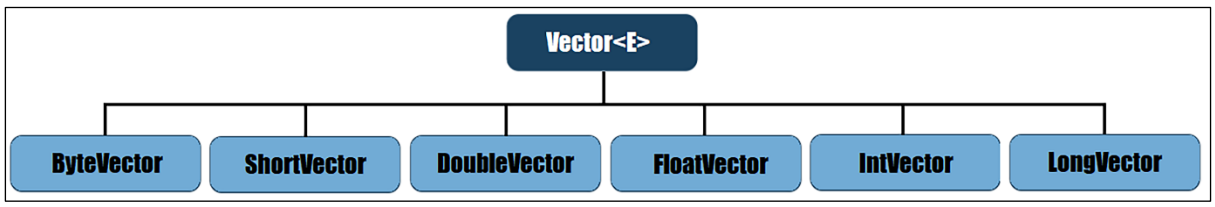
```
OBJECT_A
   class_name:MelonContainerRecord
   class_data [
    field:expiration, type:LocalDate, value:2023-02-26
    field:batch, type:String, value:ML9000SQA0
    field: melon, type:Melon, value:
      OBJECT_B
         class_name:Melon
         class_data [
          field:type, type:String, value:Gac
          field:weight, type:float, value:5000.0
         ]
    ]
```

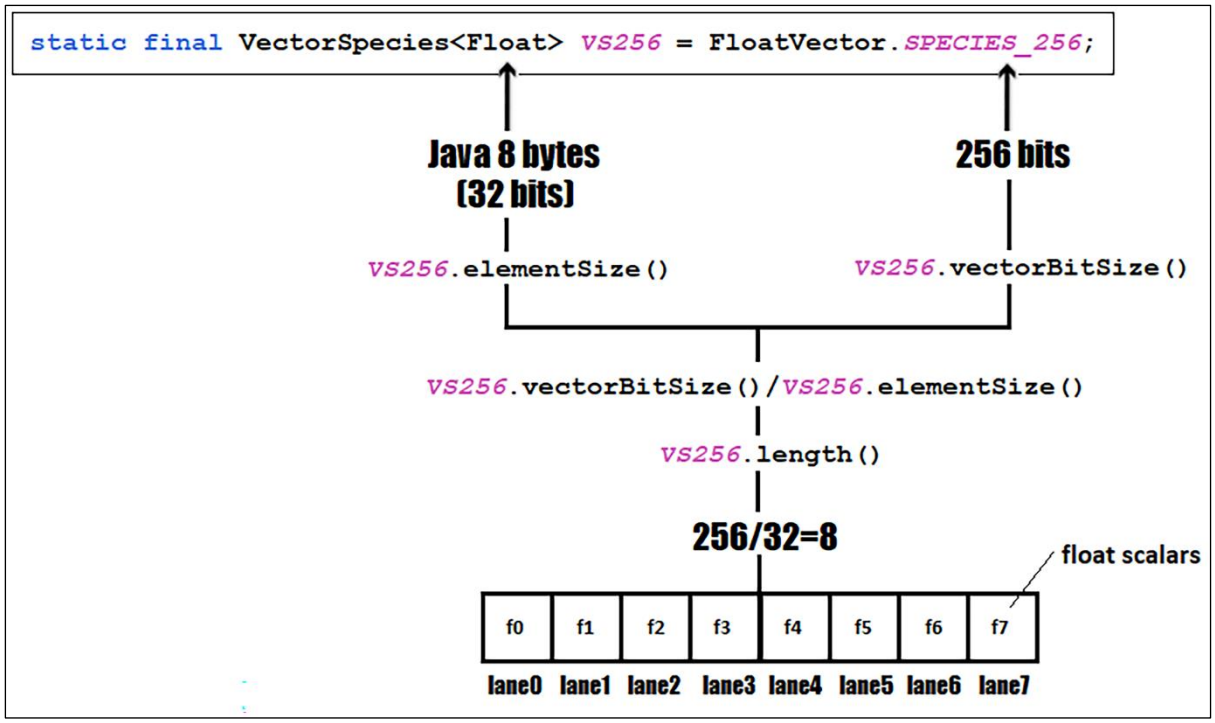
```
Exception in thread "main" java.io.InvalidObjectException: The batch format should be: MLxxxxxxxx
        at java.base/java.io.ObjectInputStream.readRecord(ObjectInputStream.java:2386)
        at java.base/java.io.ObjectInputStream.readOrdinaryObject(ObjectInputStream.java:2274)
        at java.base/java.io.ObjectInputStream.readObject0(ObjectInputStream.java:1760)
        at java.base/java.io.ObjectInputStream.readObject(ObjectInputStream.java:538)
        at java.base/java.io.ObjectInputStream.readObject(ObjectInputStream.java:496)
        at modern.challenge.Main.main(Main.java:61)
Caused by: java.lang.IllegalArgumentException: The batch format should be: MLxxxxxxxx
        at modern.challenge.MelonContainerRecord.<init>(MelonContainerRecord.java:16)
        at java.base/java.io.ObjectInputStream.readRecord(ObjectInputStream.java:2384)
        ... 5 more
```

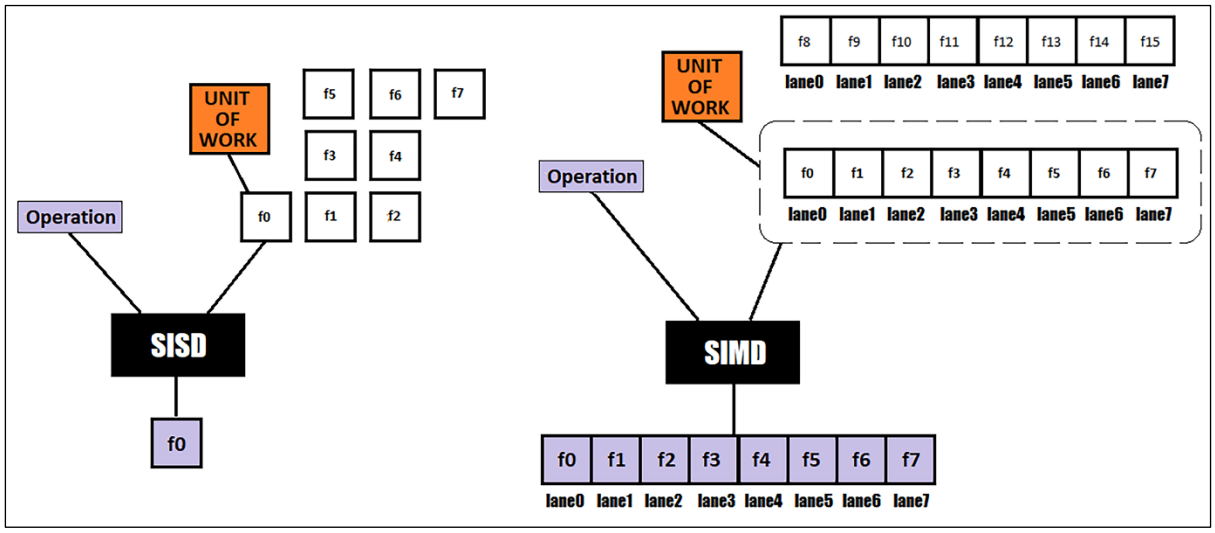
Chapter 5: Arrays, Collections, and Data Structures

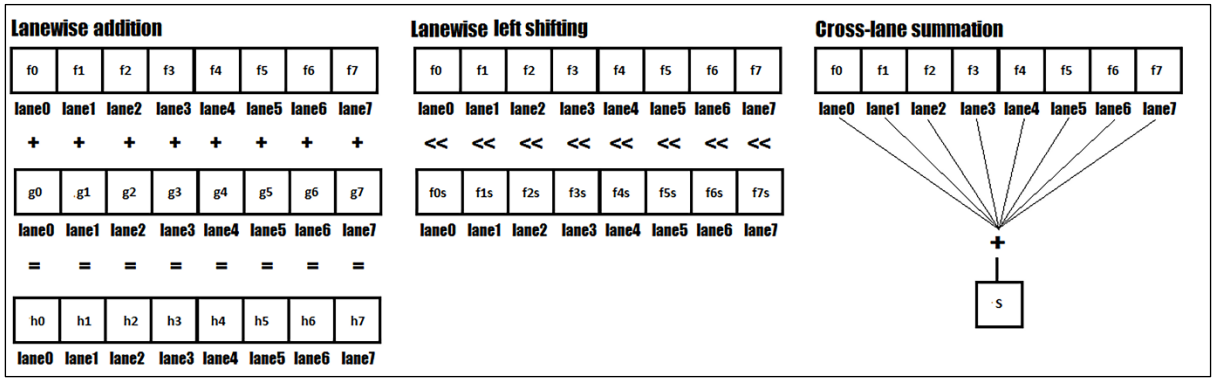


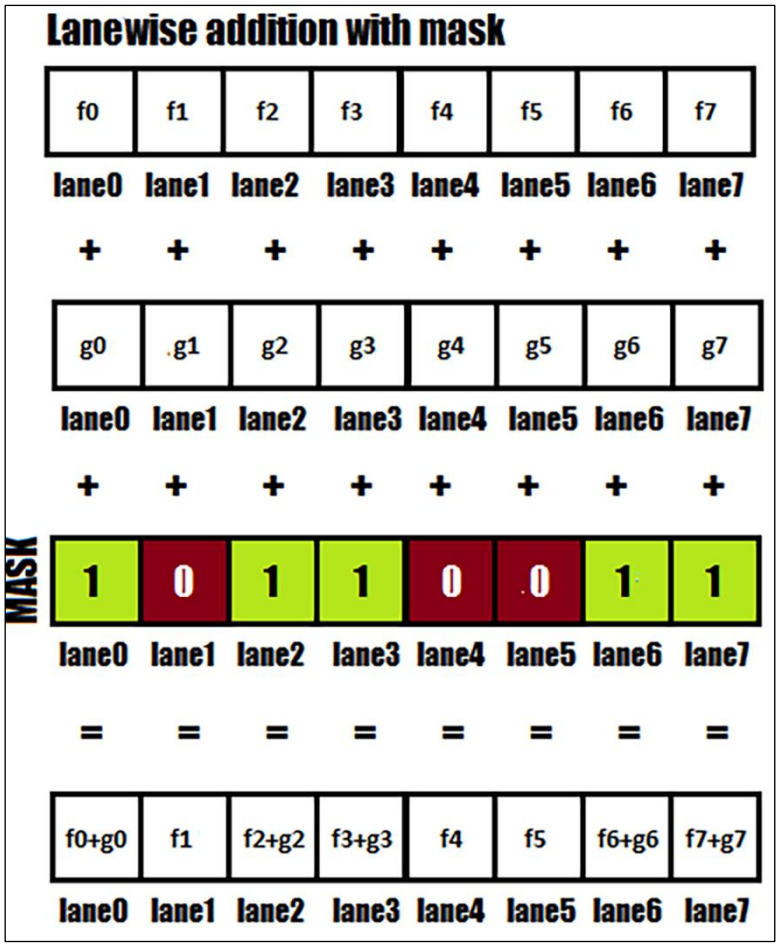


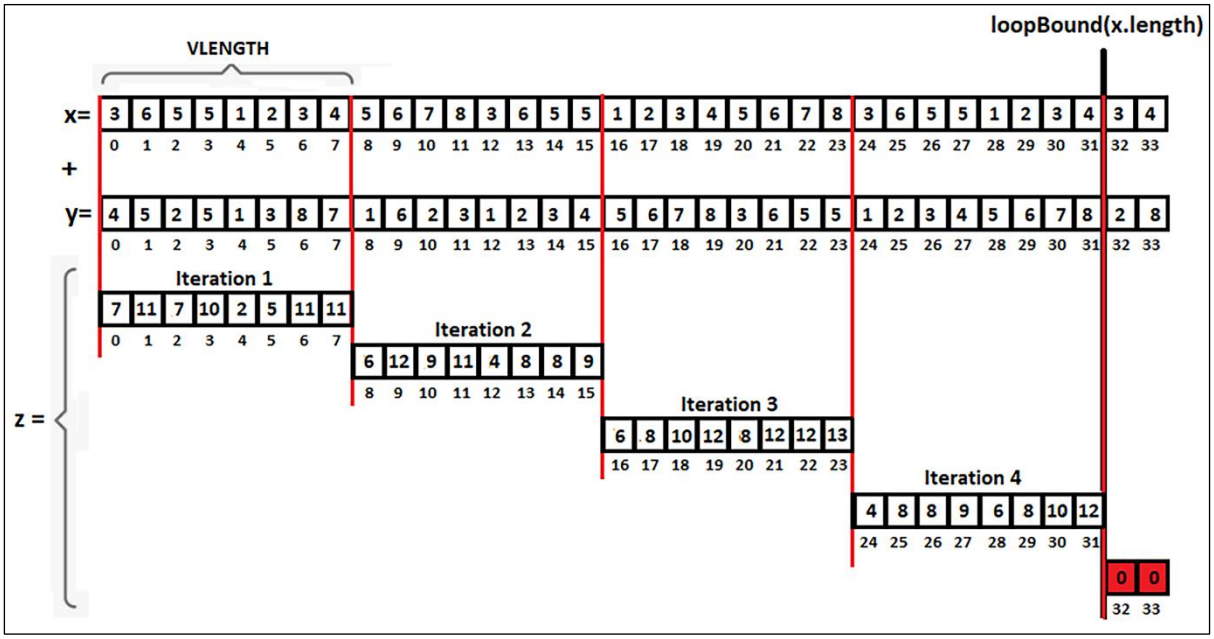



static final VectorSpecies<Float> VS256 = FloatVector.SPECIES_256;
Java 8 bytes
(32 bits)
256 bits
VS256.elementSize()
VS256.vectorBitSize()
VS256.vectorBitSize()/VS256.elementSize()
VS256.length()
256/32=8
float scalars
f0 f1 f2 f3 f4 f5 f6 f7
lane0 lane1 lane2 lane3 lane4 lane5 lane6 lane7


UNIT OF WORK
Operation
SISD
f0
f5 f6 f7
f3 f4
f0 f1 f2
UNIT OF WORK
Operation
SIMD
f8 f9 f10 f11 f12 f13 f14 f15
f0 f1 f2 f3 f4 f5 f6 f7
lane0 lane1 lane2 lane3 lane4 lane5 lane6 lane7


Lanewise addition
Lanewise left shifting
Cross-lane summation
f0 f1 f2 f3 f4 f5 f6 f7
lane0 lane1 lane2 lane3 lane4 lane5 lane6 lane7
+ + + + + + + +
g0 g1 g2 g3 g4 g5 g6 g7
= = = = = = = =
h0 h1 h2 h3 h4 h5 h6 h7
<< << << << << << << <<
f0s f1s f2s f3s f4s f5s f6s f7s
+
s

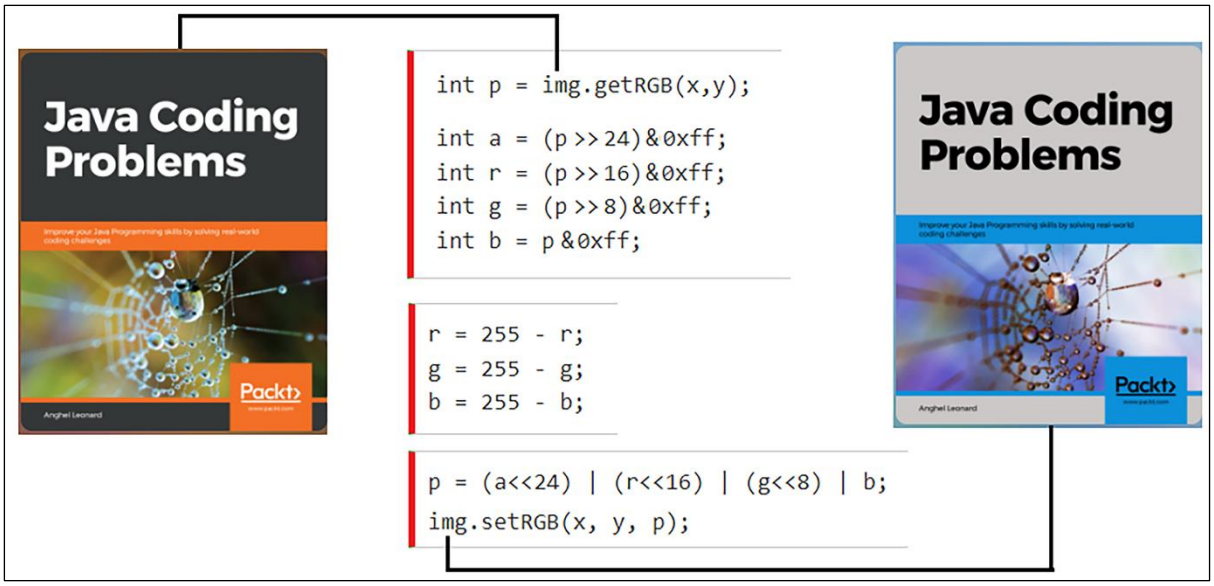

Lanewise addition with mask
f0 f1 f2 f3 f4 f5 f6 f7
lane0 lane1 lane2 lane3 lane4 lane5 lane6 lane7
+ + + + + + + +
g0 g1 g2 g3 g4 g5 g6 g7
lane0 lane1 lane2 lane3 lane4 lane5 lane6 lane7
+ + + + + + + +
MASK
1 0 1 1 0 0 1 1
lane0 lane1 lane2 lane3 lane4 lane5 lane6 lane7
= = = = = = = =
f0+g0 f1 f2+g2 f3+g3 f4 f5 f6+g6 f7+g7
lane0 lane1 lane2 lane3 lane4 lane5 lane6 lane7

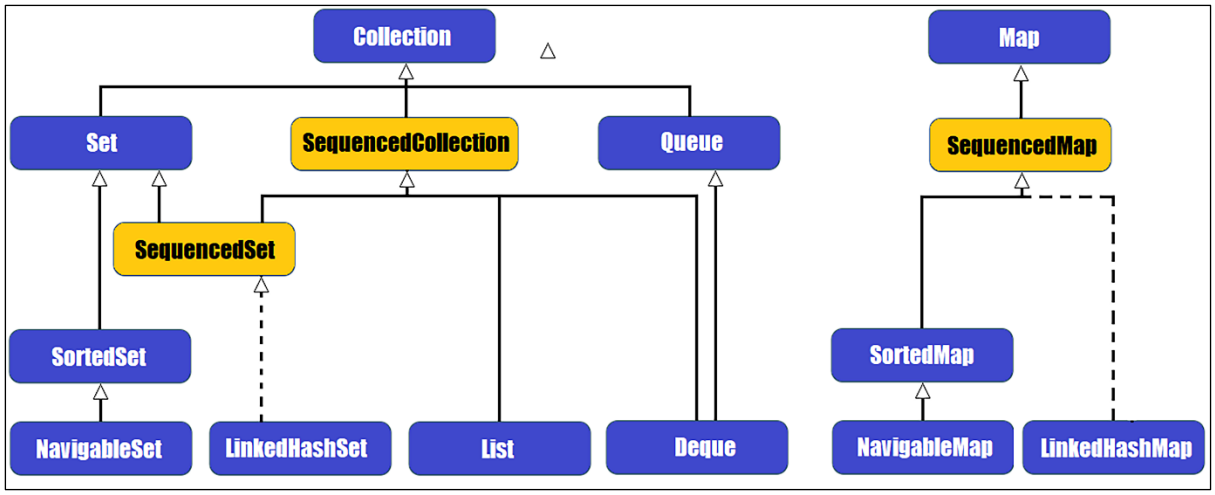

loopBound(x.length)
VLENGTH
x=
3 6 5 5 1 2 3 4 5 6 7 8 3 6 5 5 1 2 3 4 5 6 7 8 3 6 5 5 1 2 3 4 3 4
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33
+
y=
4 5 2 5 1 3 8 7 1 6 2 3 1 2 3 4 5 6 7 8 3 6 5 5 1 2 3 4 5 6 7 8 2 8
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33
Iteration 1
7 11 7 10 2 5 11 11
0 1 2 3 4 5 6 7
Iteration 2
6 12 9 11 4 8 8 9
8 9 10 11 12 13 14 15
Iteration 3
6 8 10 12 8 12 12 13
16 17 18 19 20 21 22 23
Iteration 4
4 8 8 9 6 8 10 12
24 25 26 27 28 29 30 31
0 0
32 33
z =

Benchmark	(length)	Mode	Cnt	Score	Error	Units
Main.computeArrays	50000000	thrpt	5	0.009 Â±	0.001	ops/ms
Main.computeNoMask	50000000	thrpt	5	0.019 Â±	0.001	ops/ms
Main.computeUnrolledNoMask	50000000	thrpt	5	0.019 Â±	0.001	ops/ms
Main.computeWithMask	50000000	thrpt	5	0.019 Â±	0.001	ops/ms
Main.computeArrays	50000000	avgt	5	114.407 Â±	3.544	ms/op
Main.computeNoMask	50000000	avgt	5	51.316 Â±	1.638	ms/op
Main.computeUnrolledNoMask	50000000	avgt	5	51.215 Â±	1.088	ms/op
Main.computeWithMask	50000000	avgt	5	51.667 Â±	0.736	ms/op

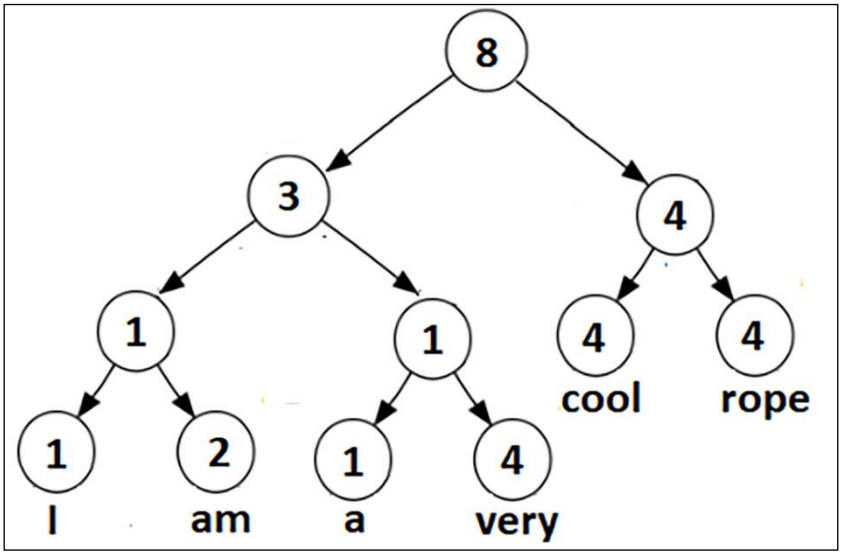
$$\begin{pmatrix} 1 & 2 & 5 & 4 \\ 5 & 4 & 3 & 7 \\ 2 & 2 & 6 & 2 \\ 4 & 1 & 2 & 5 \end{pmatrix} \cdot \begin{pmatrix} 3 & 4 & 1 & 8 \\ 7 & 4 & 9 & 9 \\ 5 & 6 & 7 & 3 \\ 5 & 8 & 7 & 7 \end{pmatrix} = \begin{pmatrix} 62 & 74 & 82 & 69 \\ 93 & 110 & 111 & 134 \\ 60 & 68 & 76 & 66 \\ 54 & 72 & 62 & 82 \end{pmatrix}$$

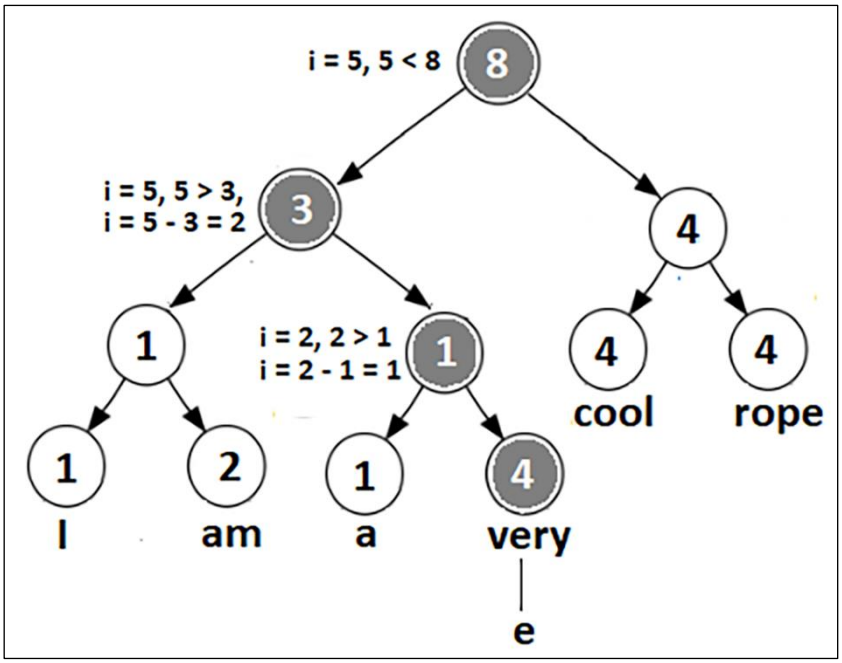
X Y Z

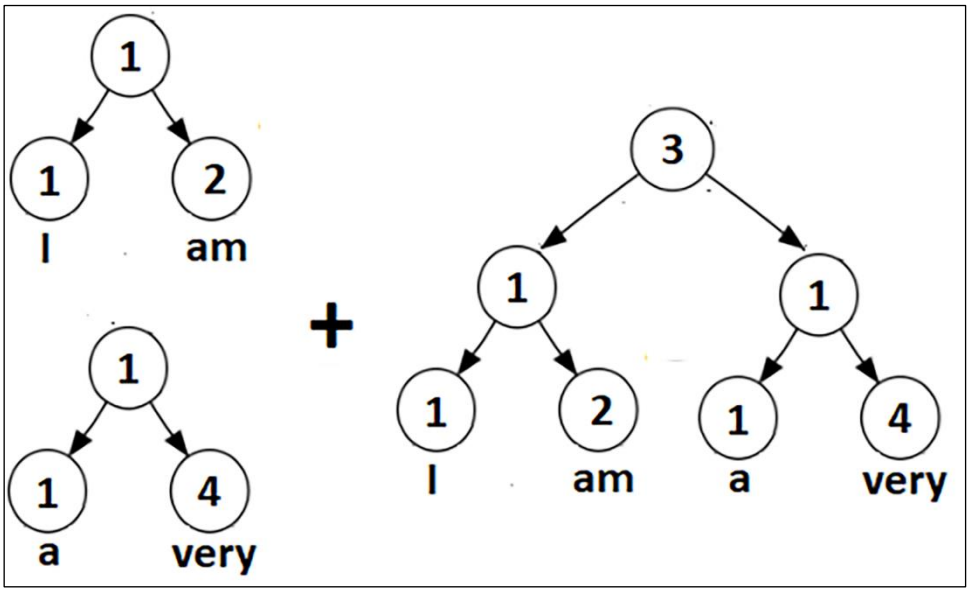




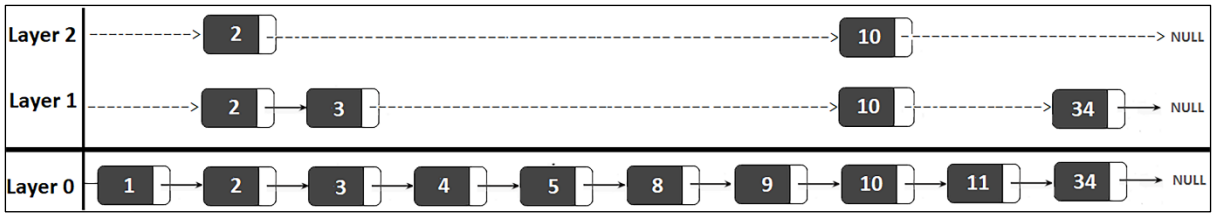
Operation	Rope	String
Index	O(log n)	O(1)
Insert	O(log n) no rebalancing O(n) worst case	O(n)
Delete	O(log n)	O(n)
Concat	O(log n) no rebalancing O(n) worst case	O(n)
Split	O(log n)	O(1)

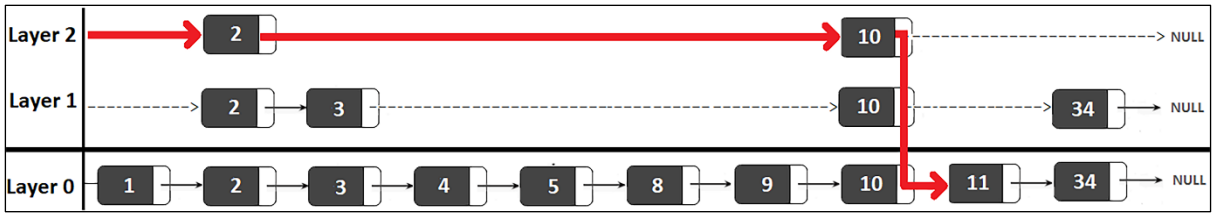


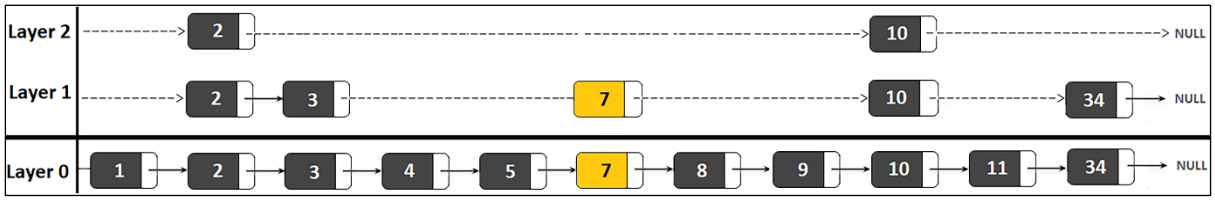


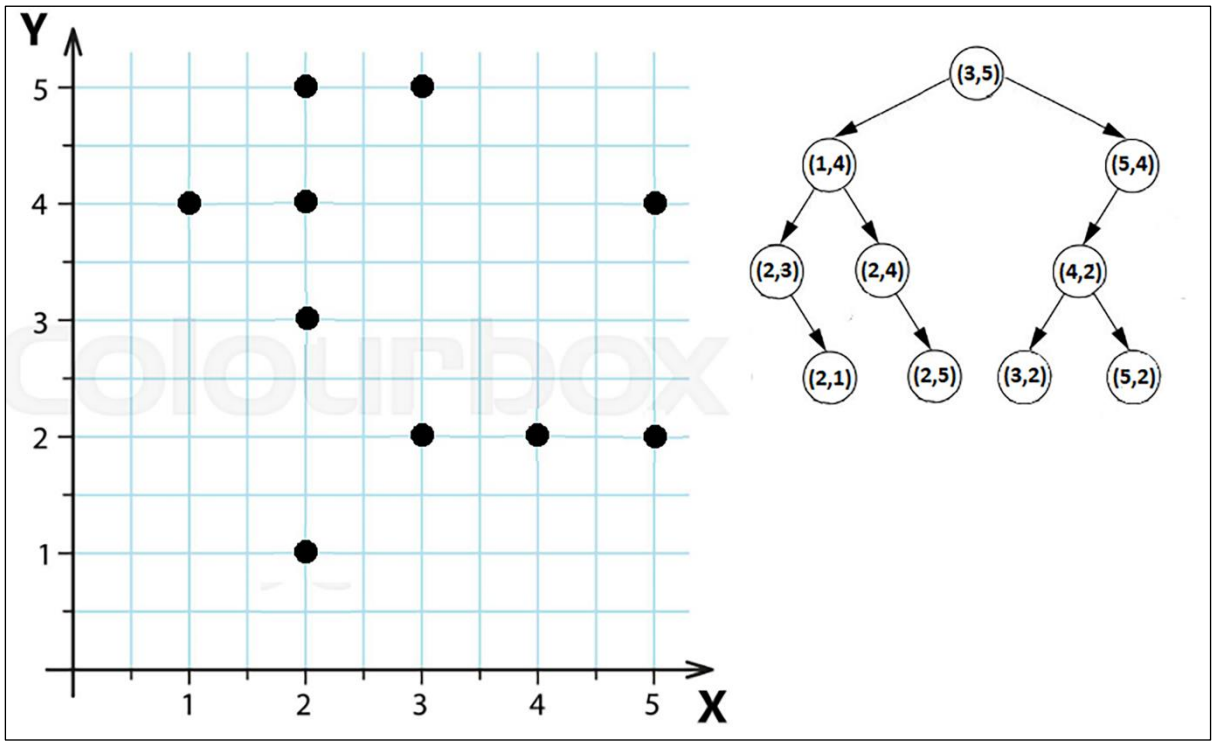


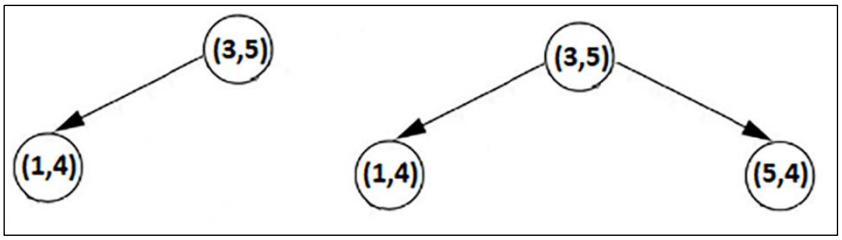
Operation	Average	Worst
Search	O(log n)	O(n)
Insert	O(log n)	O(n)
Delete	O(log n)	O(n)

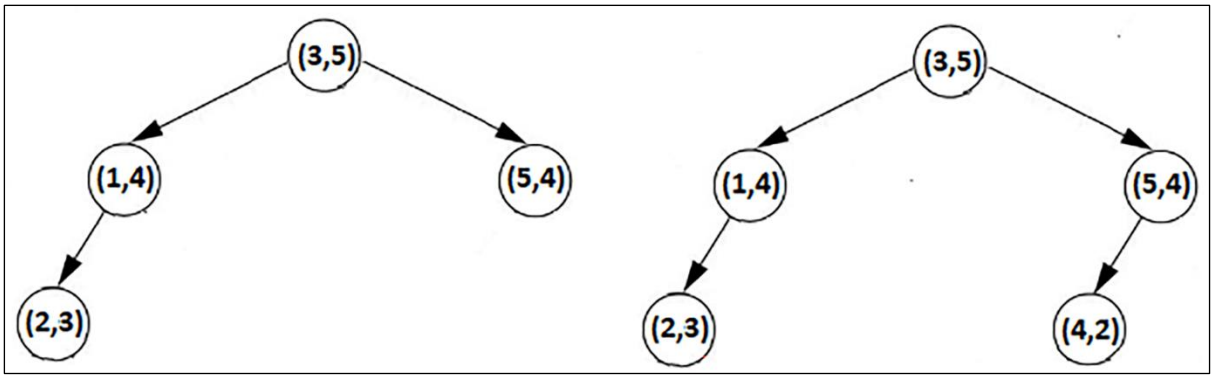

Layer 2
2
10
NULL
Layer 1
2
3
10
34
NULL
Layer 0
1
2
3
4
5
8
9
10
11
34
NULL

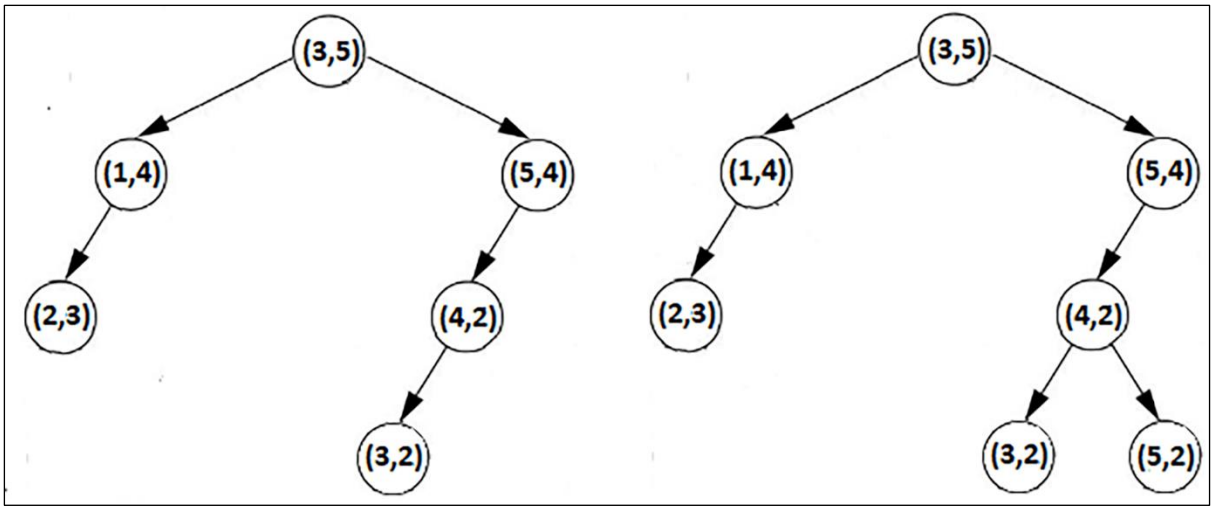

Layer 2
2
10
NULL
Layer 1
2
3
10
34
NULL
Layer 0
1
2
3
4
5
8
9
10
11
34
NULL

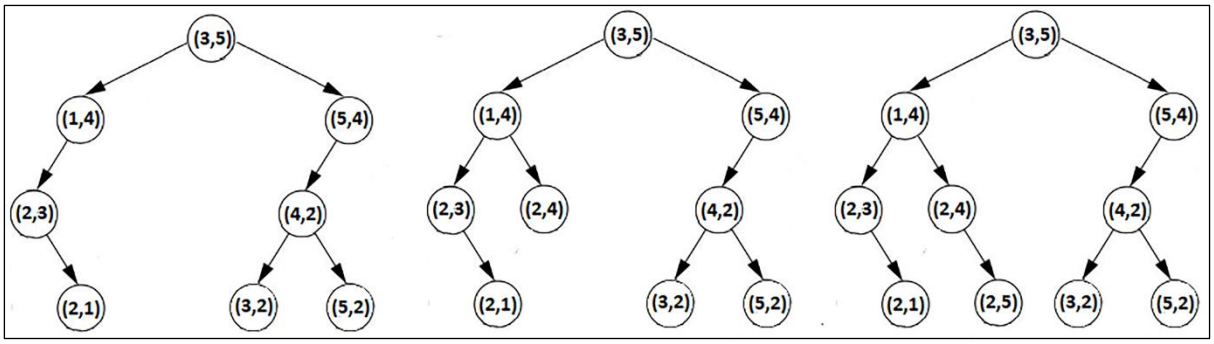

Layer 2
2
10
NULL
Layer 1
2
3
7
10
34
NULL
Layer 0
1
2
3
4
5
7
8
9
10
11
34
NULL

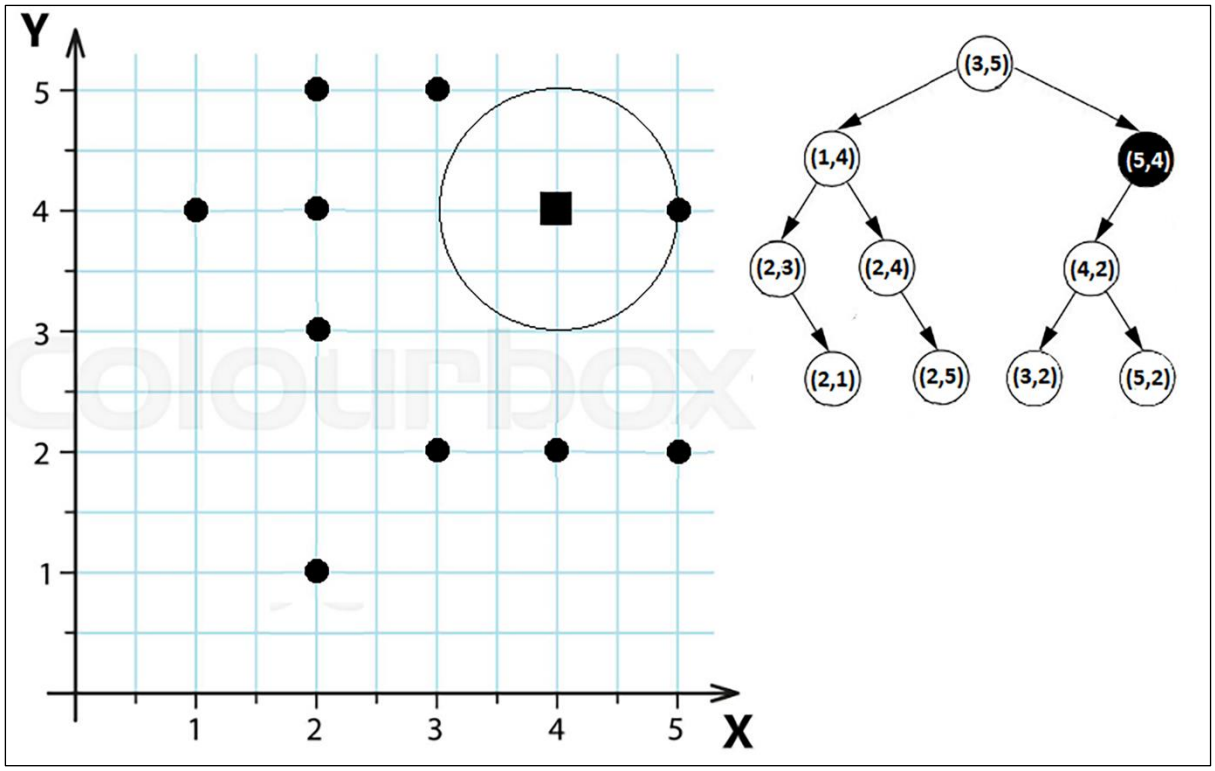

Y
5
4
3
2
1
1
2
3
4
5
X
(3,5)
(1,4)
(5,4)
(2,3)
(2,4)
(4,2)
(2,1)
(2,5)
(3,2)
(5,2)

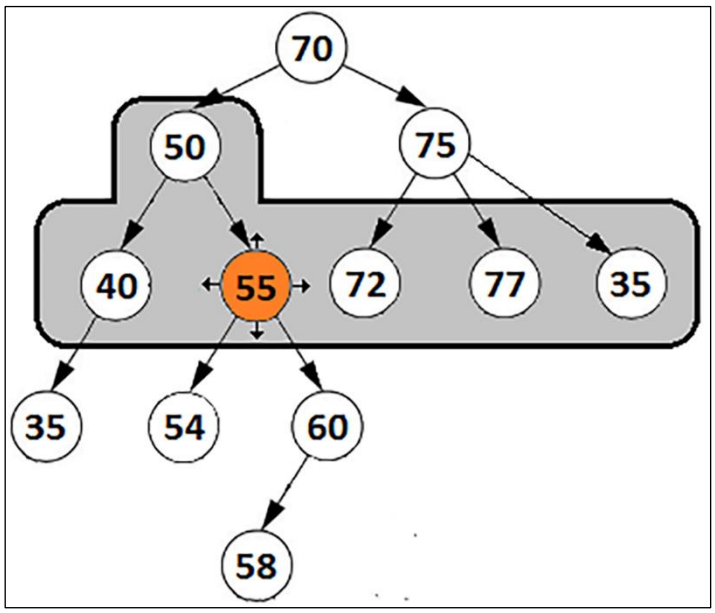

(3,5)
(1,4)
(3,5)
(1,4)
(5,4)

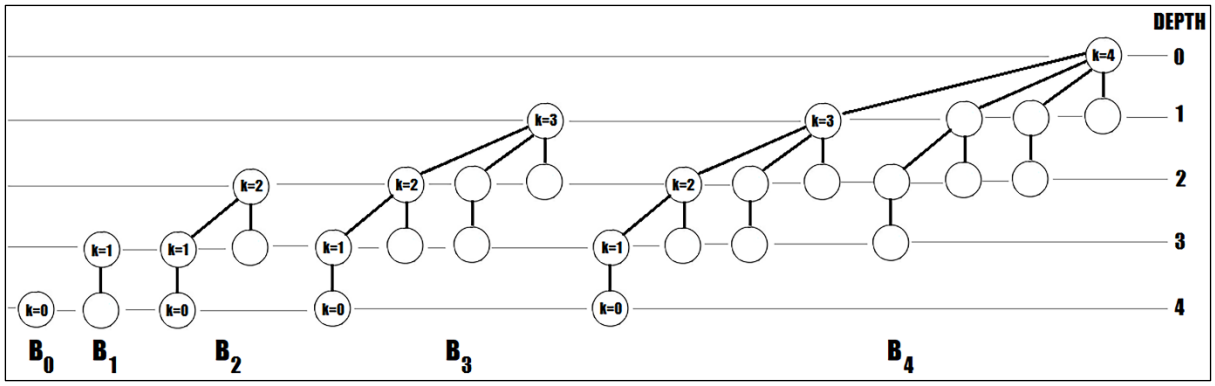

(3,5)
(1,4)
(5,4)
(2,3)
(3,5)
(1,4)
(5,4)
(2,3)
(4,2)


(3,5)
(1,4)
(5,4)
(2,3)
(4,2)
(3,2)
(3,5)
(1,4)
(5,4)
(2,3)
(4,2)
(3,2)
(5,2)

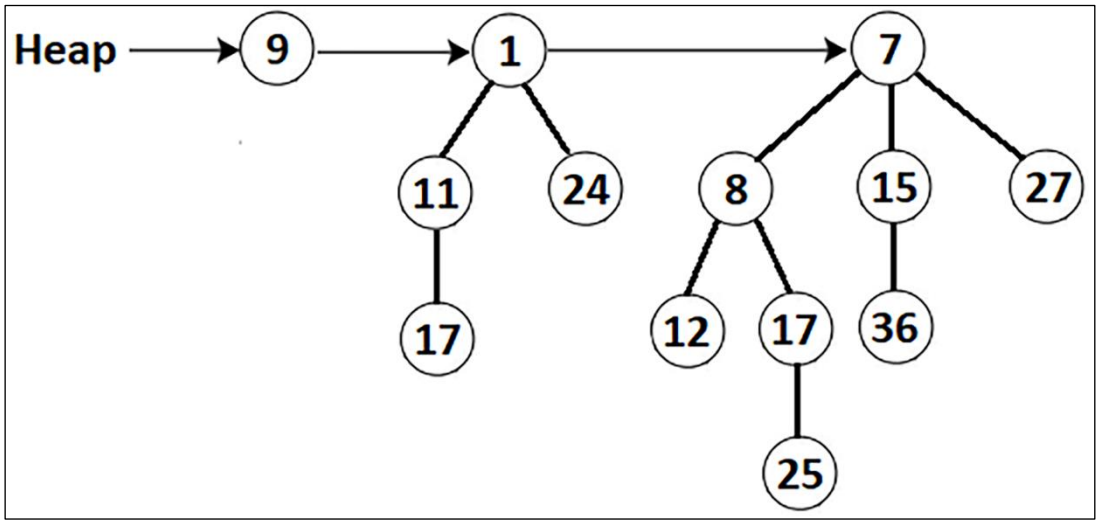

(3,5)
(1,4)
(5,4)
(2,3)
(4,2)
(2,1)
(3,2)
(5,2)
(3,5)
(1,4)
(5,4)
(2,3)
(2,4)
(4,2)
(2,1)
(3,2)
(5,2)
(3,5)
(1,4)
(5,4)
(2,3)
(2,4)
(4,2)
(2,1)
(2,5)
(3,2)
(5,2)

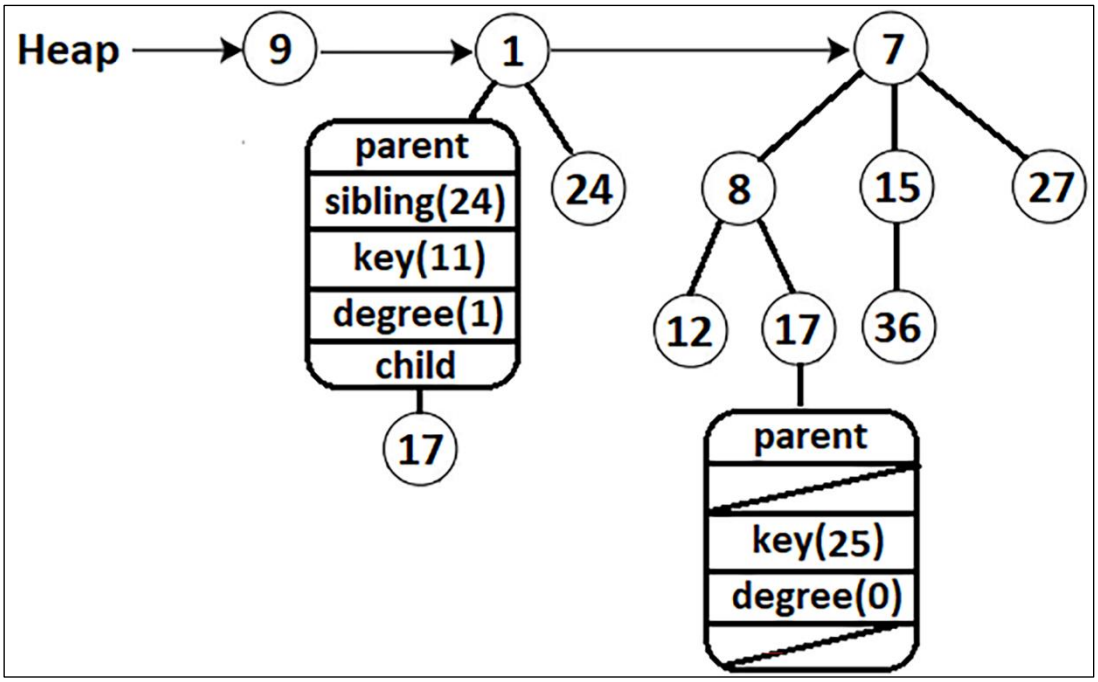

Y
X
1
2
3
4
5
(3,5)
(1,4)
(5,4)
(2,3)
(2,4)
(4,2)
(2,1)
(2,5)
(3,2)
(5,2)

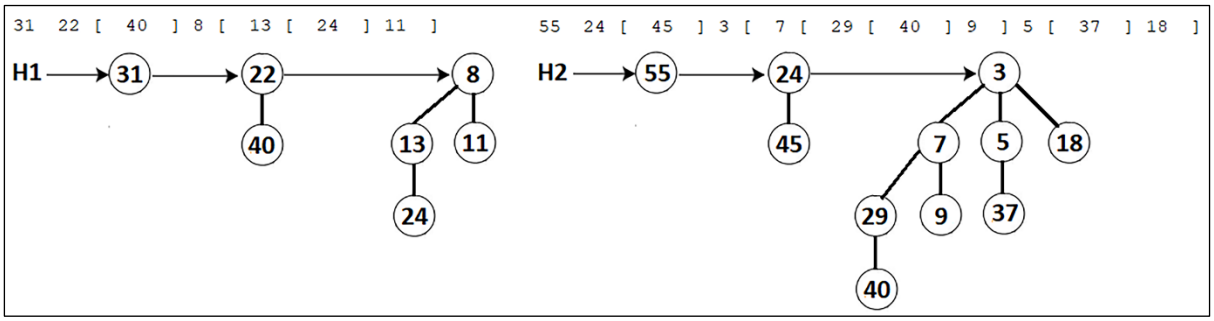

70
50
75
40
55
72
77
35
35
54
60
58

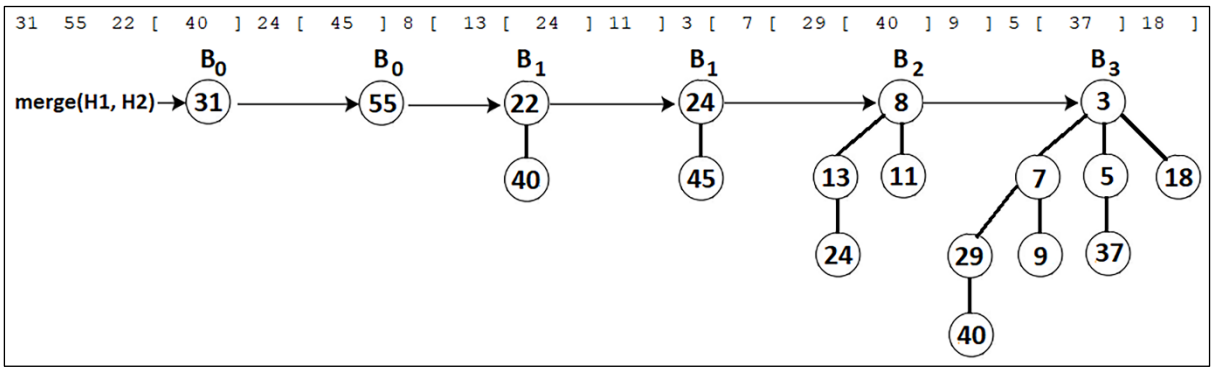


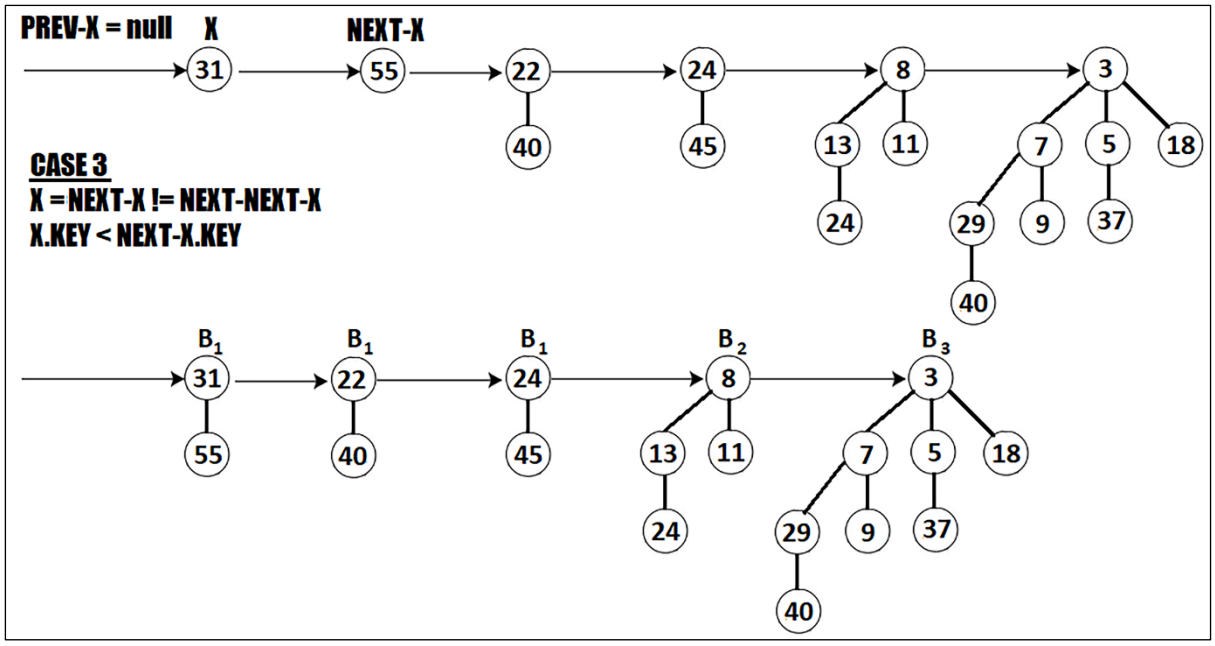
Operation	Worst
Insert	O(log n)
Find minimum key	O(log n)
Extract minimum key	O(log n)
Decreasing a key	O(log n)
Deleting a node	O(log n)
Union or merging	O(log n)

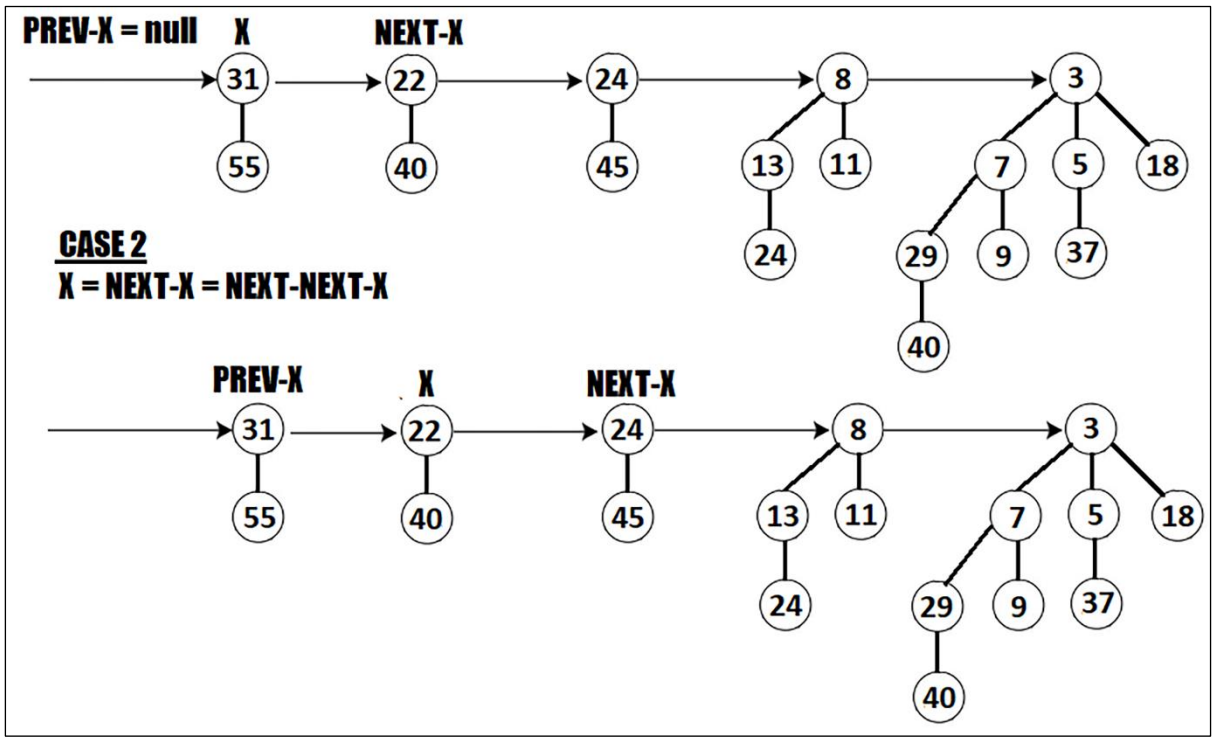


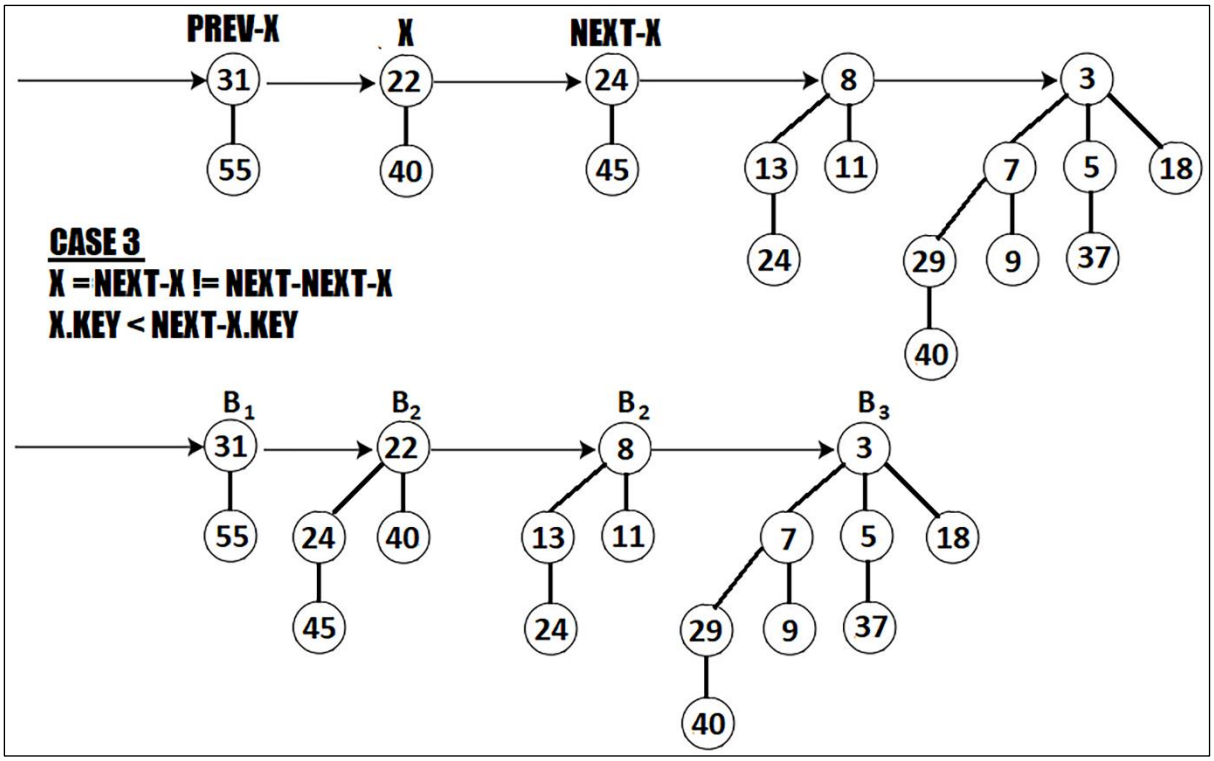

Heap
9
1
7
parent
sibling(24)
key(11)
degree(1)
child
17
24
8
15
27
12
17
36
parent
key(25)
degree(0)

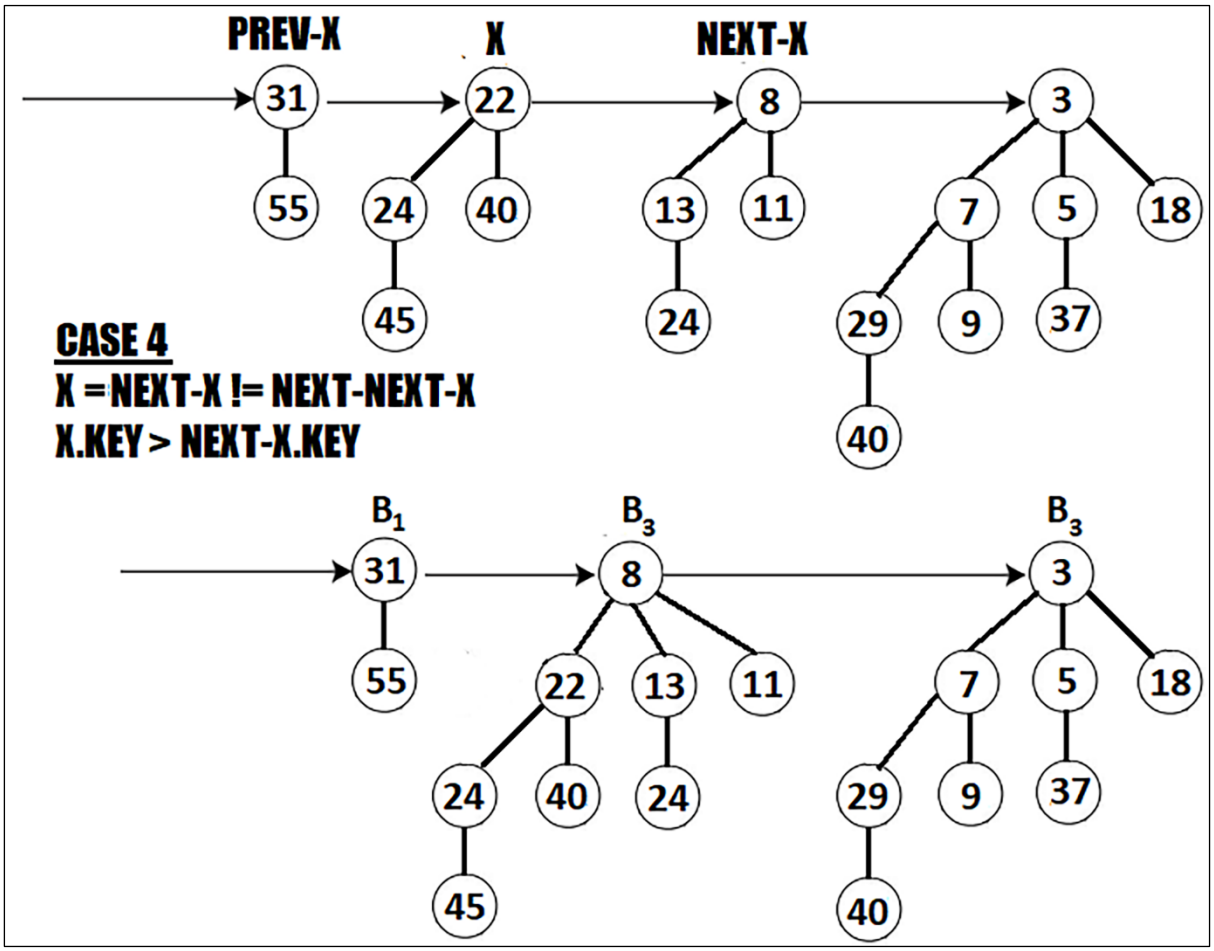

31 22 [40] 8 [13 [24] 11]
H1
31
22
8
40
13
11
24
55 24 [45] 3 [7 [29 [40] 9] 5 [37] 18]
H2
55
24
3
45
7
5
18
29
9
37
40

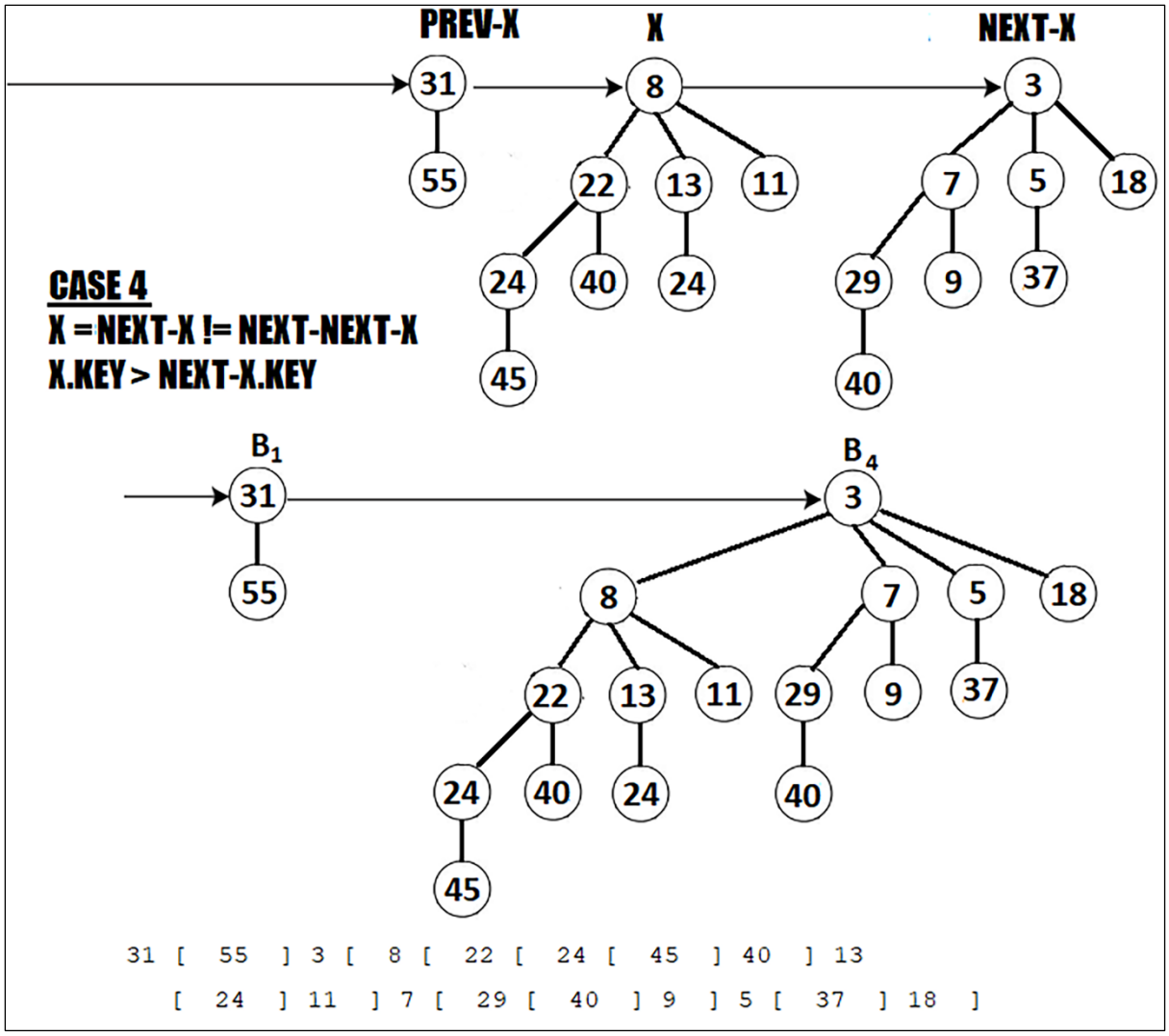

31 55 22 [40] 24 [45] 8 [13 [24] 11] 3 [7 [29 [40] 9] 5 [37] 18]
merge(H1, H2)
B0
B0
B1
B1
B2
B3
31
55
22
24
8
3
40
45
13
11
7
5
18
24
29
9
37
40

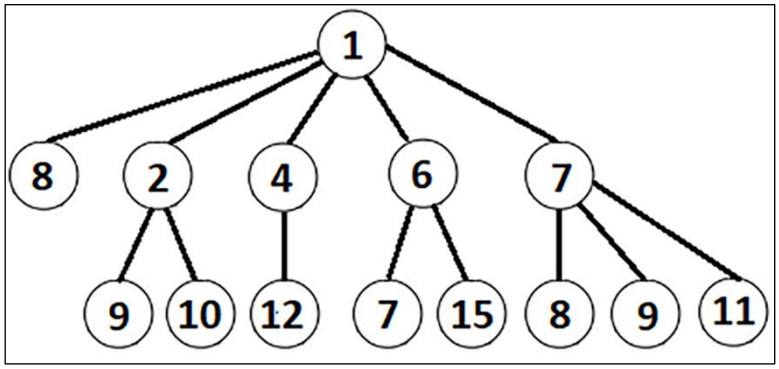

PREV-X = null
X
NEXT-X
31
55
22
40
24
45
8
13
11
24
3
7
5
18
29
9
37
40
CASE 3
X = NEXT-X != NEXT-NEXT-X
X.KEY < NEXT-X.KEY
B1
B1
B1
B2
B3

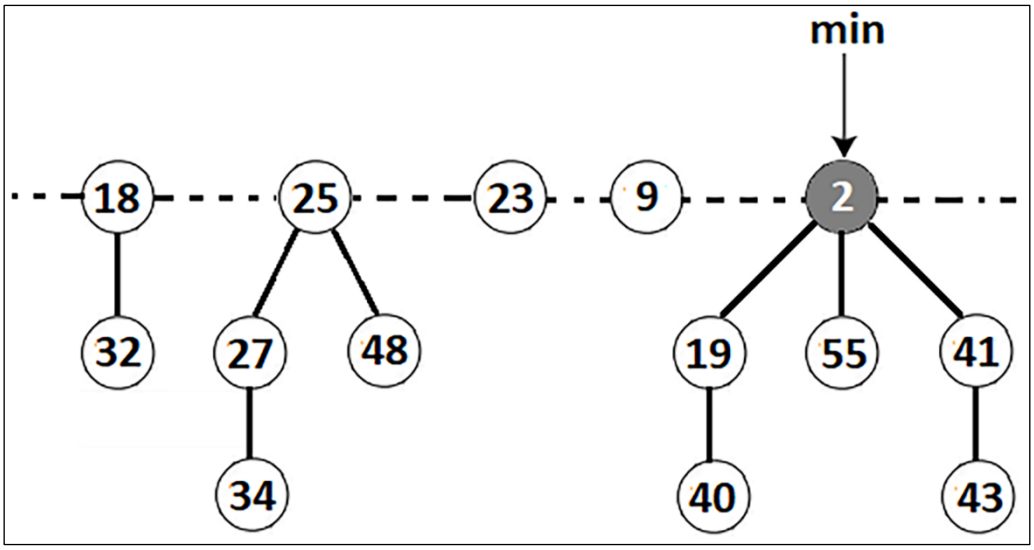

PREV-X = null
X
NEXT-X
31
55
22
40
24
45
8
13
11
24
3
7
5
18
29
9
37
40
CASE 2
X = NEXT-X = NEXT-NEXT-X
PREV-X
X
NEXT-X

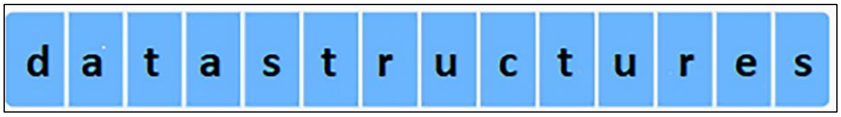

PREV-X
X
NEXT-X
31
22
24
8
3
55
40
45
13
11
7
5
18
24
29
9
37
40
CASE 3
X = NEXT-X != NEXT-NEXT-X
X.KEY < NEXT-X.KEY
B1
B2
B2
B3
31
22
8
3
55
24
40
13
11
7
5
18
45
24
29
9
37
40


PREV-X
X
NEXT-X
31
22
8
3
55
24
40
13
11
7
5
18
45
24
29
9
37
40
CASE 4
X = NEXT-X != NEXT-NEXT-X
X.KEY > NEXT-X.KEY
B1
B3
B3
31
8
3
55
22
13
11
7
5
18
24
40
24
29
9
37
45
40

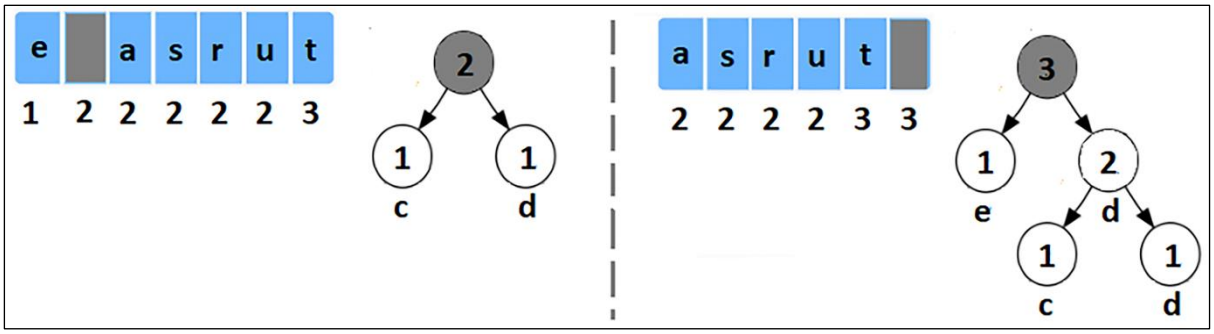

PREV-X
X
NEXT-X
CASE 4
X = NEXT-X != NEXT-NEXT-X
X.KEY > NEXT-X.KEY
B1
B4
31 [55] 3 [8 [22 [24 [45] 40] 13
[24] 11] 7 [29 [40] 9] 5 [37] 18]

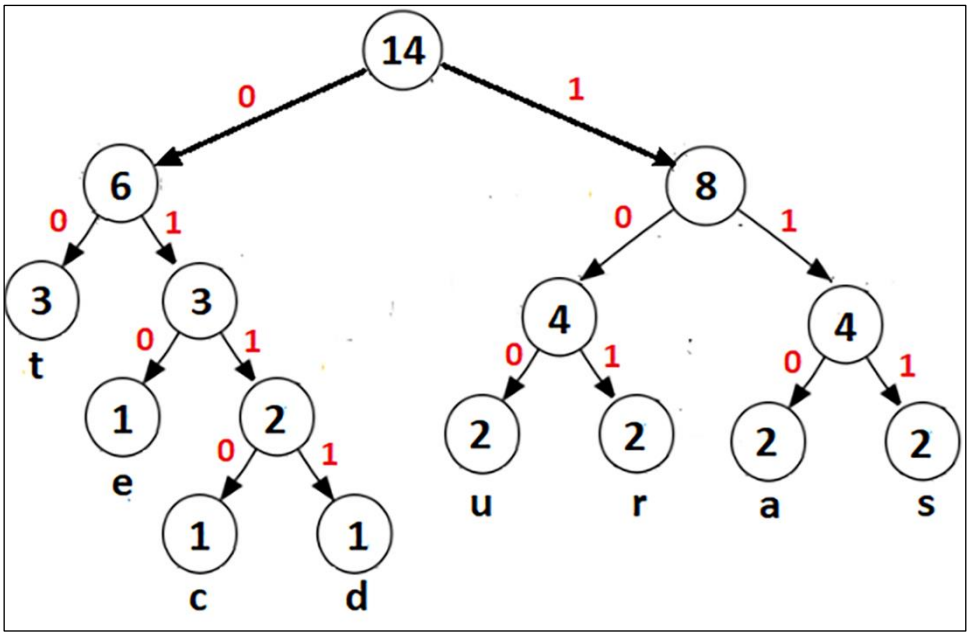



min
18
25
23
9
2
32
27
48
19
55
41
34
40
43

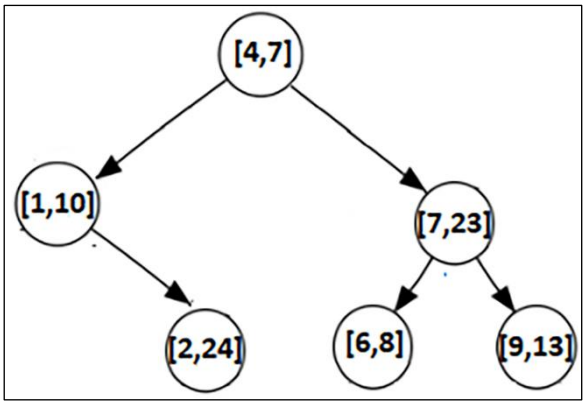

d a t a s t r u c t u r e s

d c e a s r u t
1 1 1 2 2 2 2 3


e a s r u t
1 2 2 2 2 2 3
2
1
1
c
d
a s r u t
2 2 2 2 3 3
3
1
2
e
d
1
1
c
d



Character	Frequency	Code	Bits
a	2	110	6
c	1	0110	4
d	1	0111	4
e	1	010	3
r	2	101	6
s	2	111	6
t	3	00	6
u	2	100	6
			41



The author table

authorId	name
1	Author_1
2	Author_2
3	Author_3
4	Author_4
5	Author_5

JOIN

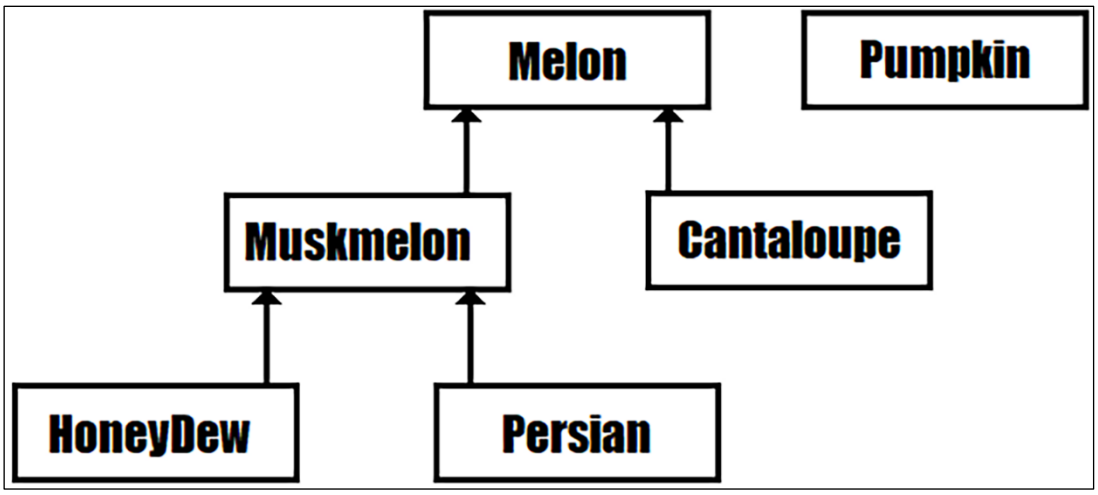
The book table

bookId	title	authorId
1	Book_1	1
2	Book_2	1
3	Book_3	2
4	Book_4	3
5	Book_5	3
6	Book_6	3
7	Book_7	4
8	Book_8	5
9	Book_9	5

author JOIN book

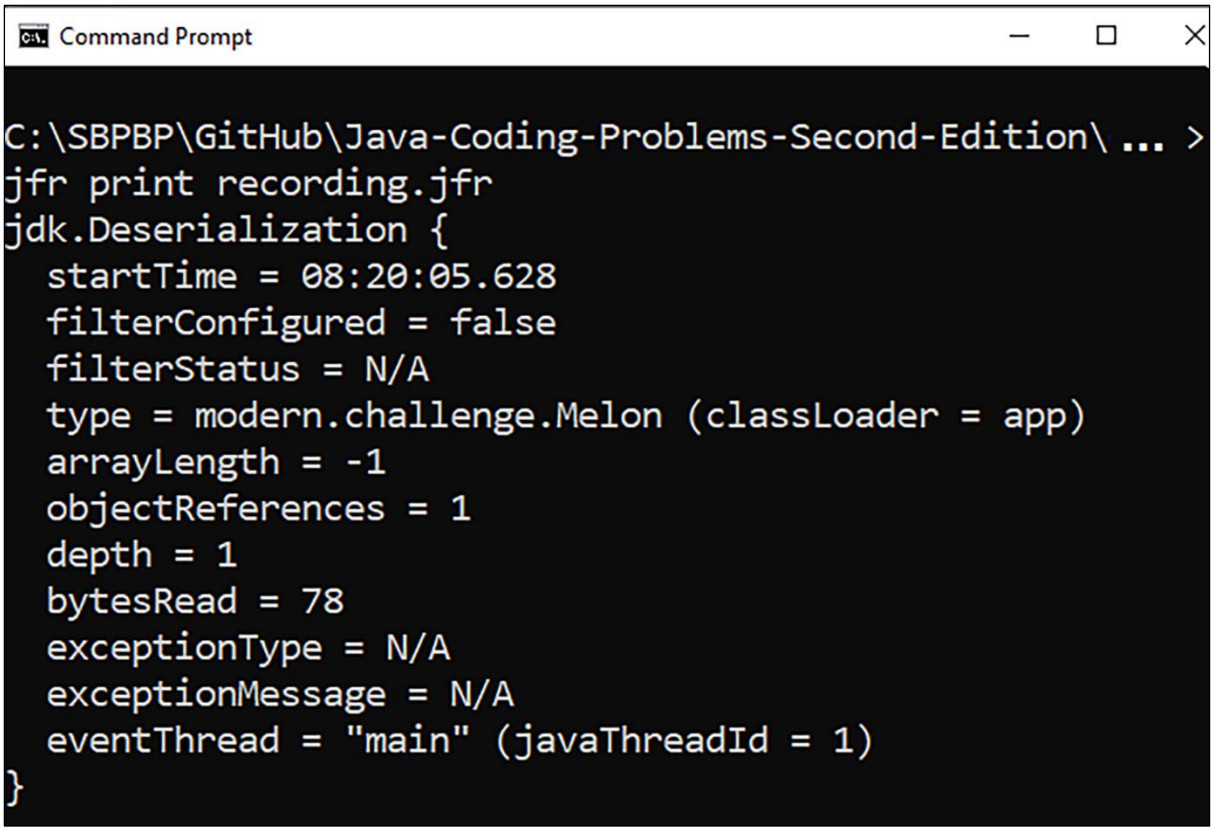
authorId	name	title	bookId
1	Author_1	Book_1	1
1	Author_1	Book_2	2
2	Author_2	Book_3	3
3	Author_3	Book_4	4
3	Author_3	Book_5	5
3	Author_3	Book_6	6
4	Author_4	Book_7	7
5	Author_5	Book_8	8
5	Author_5	Book_9	9

Chapter 6: Java I/O: Context-Specific Deserialization Filters




```
Command Prompt
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter06\P143_MonitoringDeserializationJfr>
java -XX:StartFlightRecording=filename=recording.jfr,settings=deserializationEvent.jfc -classpat
h target\classes modern.challenge.Main
```

```
Command Prompt
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter06\P143_MonitoringDeserializationJfr>
java -XX:StartFlightRecording=filename=recording.jfr,settings=deserializationEvent.jfc -classpat
h target\classes modern.challenge.Main
[0.703s][info][jfr,startup] Started recording 1. No limit specified, using maxsize=250MB as defa
ult.
[0.703s][info][jfr,startup]
[0.703s][info][jfr,startup] Use jcmd 13728 JFR.dump name=1 to copy recording data to file.
Serialization: [-84, -19, 0, 5, 115, 114, 0, 22, 109, 111, 100, 101, 114, 110, 46, 99, 104, 97,
108, 108, 101, 110, 103, 101, 46, 77, 101, 108, 111, 110, -39, 106, -25, 24, 13, -116, -59, -98,
 2, 0, 2, 70, 0, 6, 119, 101, 105, 103, 104, 116, 76, 0, 4, 116, 121, 112, 101, 116, 0, 18, 76,
106, 97, 118, 97, 47, 108, 97, 110, 103, 47, 83, 116, 114, 105, 110, 103, 59, 120, 112, 69, 28,
64, 0, 116, 0, 3, 71, 97, 99]

Deserialization: Melon{type=Gac, weight=2500.0}
```


Command Prompt
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\ ... >
jfr print recording.jfr
jdk.Deserialization {
startTime = 08:20:05.628
filterConfigured = false
filterStatus = N/A
type = modern.challenge.Melon (classLoader = app)
arrayLength = -1
objectReferences = 1
depth = 1
bytesRead = 78
exceptionType = N/A
exceptionMessage = N/A
eventThread = "main" (javaThreadId = 1)
}

Chapter 7: Foreign (Function) Memory API

```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter07\P144_EngagingJNI>
  javac -h src/main/java/modern/challenge/cpp -d target/classes src/main/java/modern/challenge/Main.java
```

```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter07\P144_EngagingJNI>
    g++ -c "-I%JAVA_HOME%\include" "-I%JAVA_HOME%\include\win32"

    src/main/java/modern/challenge/cpp/modern_challenge_Main.cpp

    -o jni/cpp/modern_challenge_Main.o
```

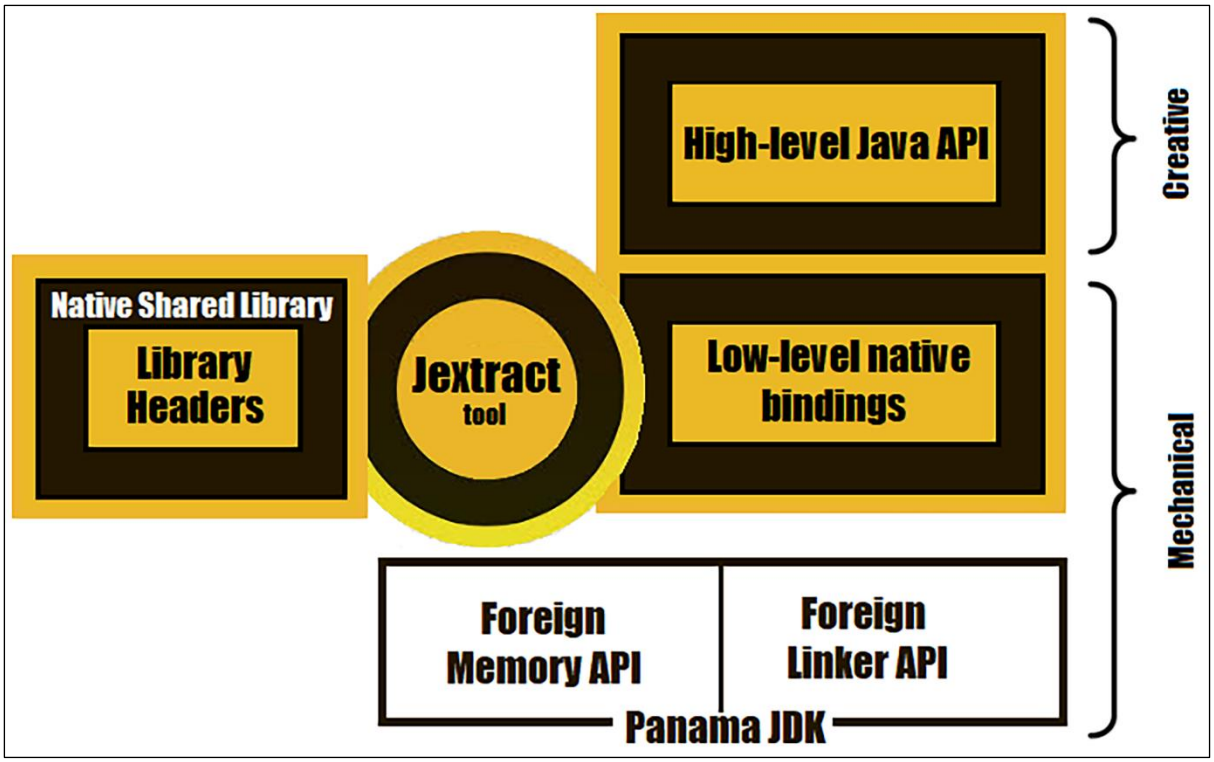
```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter07\P144_EngagingJNI>
    g++ -shared -o jni/cpp/math.dll jni/cpp/modern_challenge_Main.o

    -static -m64 -Wl,--add-stdcall-alias
```

```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter07\P144_EngagingJNI>
   java -Djava.library.path=jni/cpp -classpath target/classes modern.challenge.Main


    C++: The received arguments are : 3 and 9
    Result: 12
```

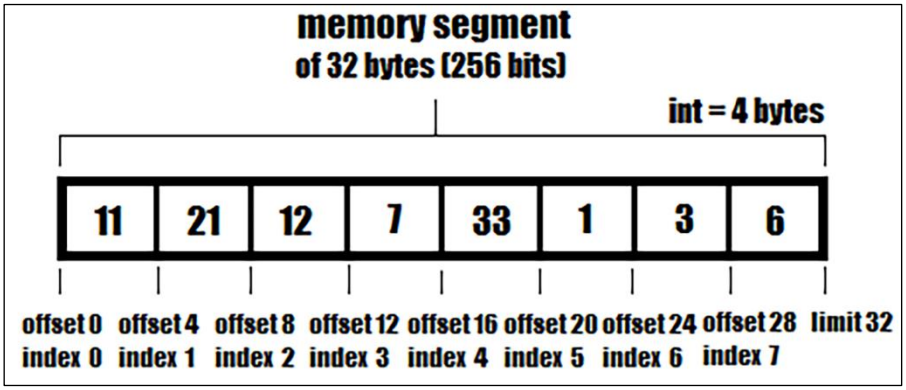
```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter07\P145_EngagingJNA>
    g++ -c "-I%JAVA_HOME%\include" "-I%JAVA_HOME%\include\win32"

    src/main/java/modern/challenge/cpp/Arithmetic.cpp

    -o jna/cpp/Arithmetic.o
```

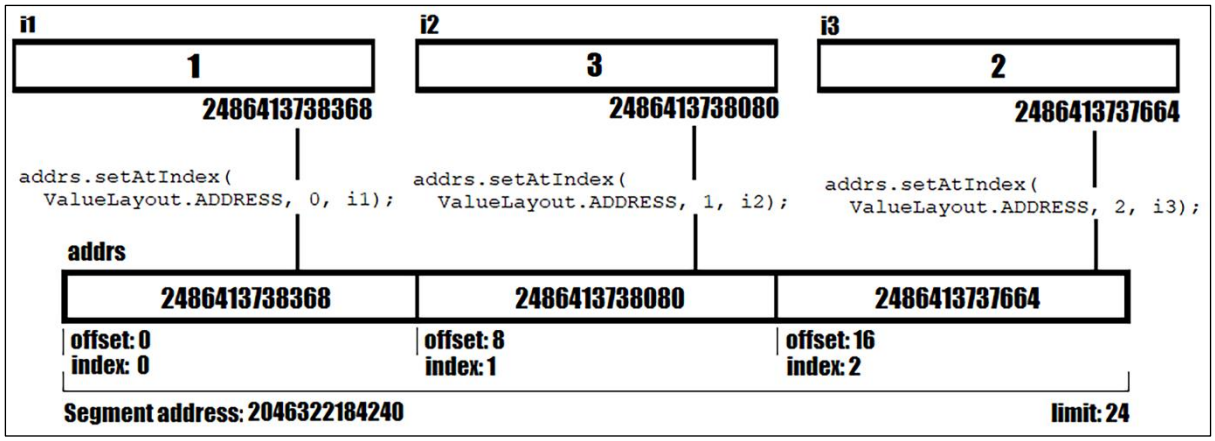
```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter07\P145_EngagingJNA>
   g++ -shared -o jna/cpp/math.dll jna/cpp/Arithmetic.o -static -m64 -Wl,--add-stdcall-alias
```

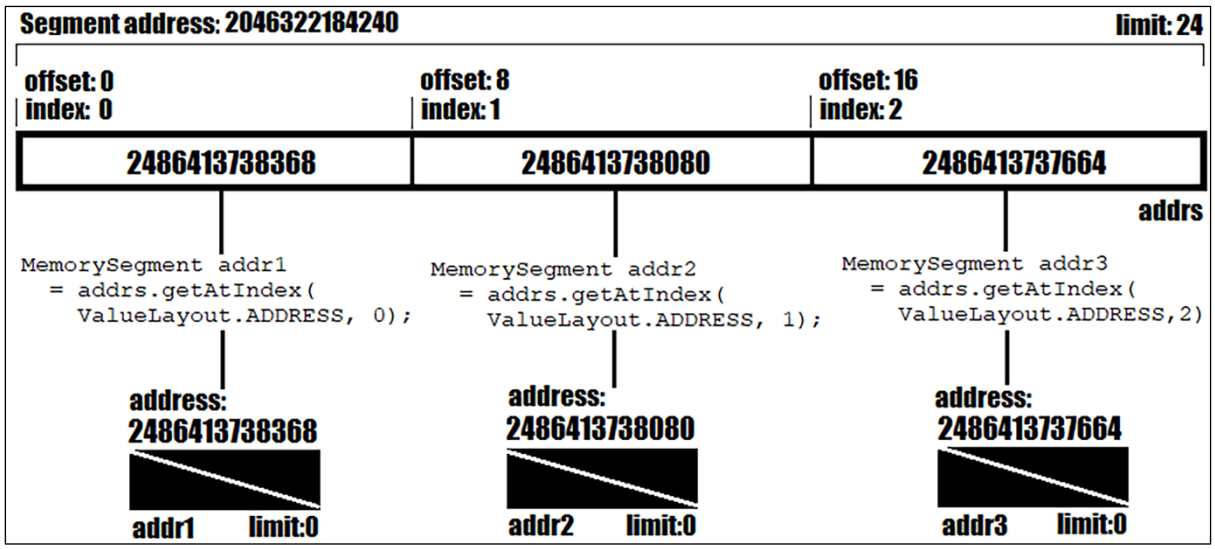
IC	18 (0x0012)	N/A	_Unwind_SetIP	0x0000bf90	None
IC	19 (0x0013)	N/A	_Z9sumTwoIntii	0x000013b0	Demumble
IC	20 (0x0014)	N/A	__emutls_get_address	0x0000c570	None

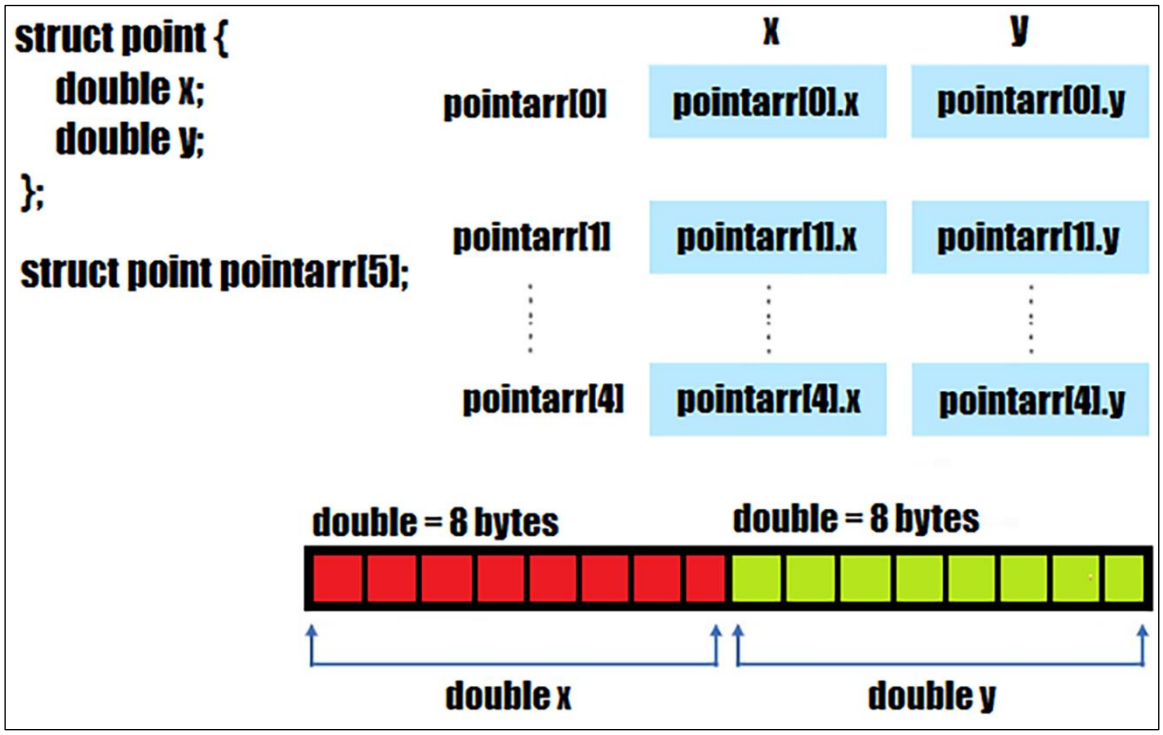


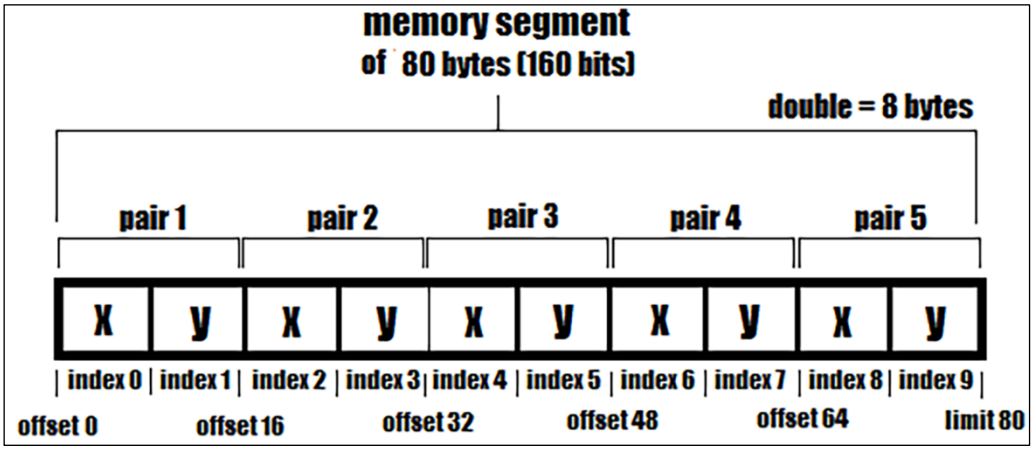
Arena's scope	Bounded lifespan	Explicitly closeable	Multiple threads
Arena.global()	no	no	yes
Arena.ofAuto()	yes	no	yes
Arena.ofConfined()	yes	yes	no
Arena.ofShared()	yes	yes	yes

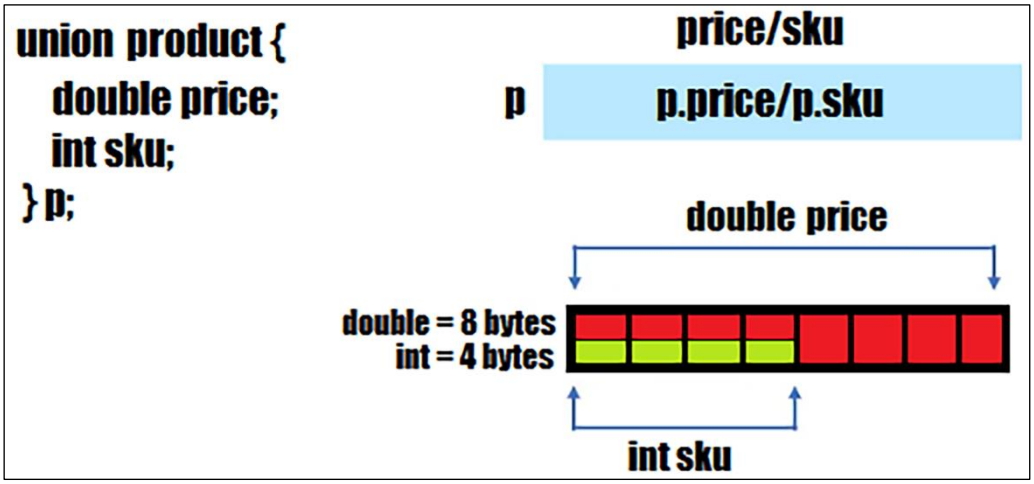


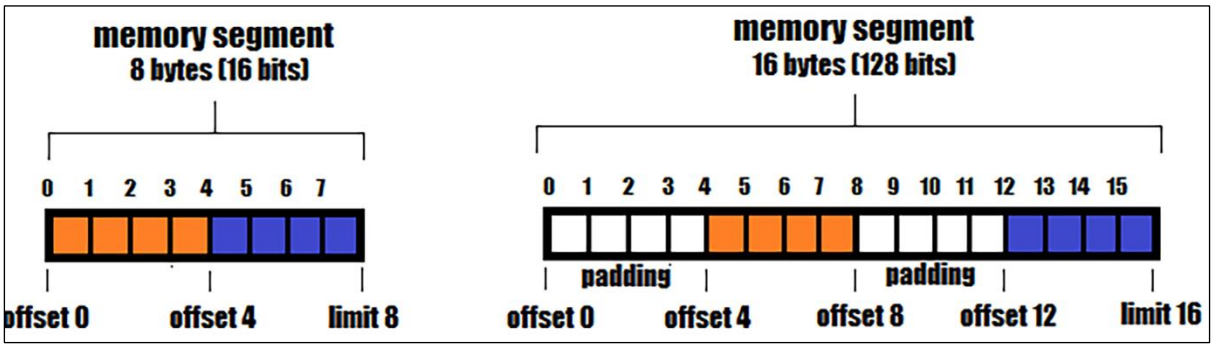

i1
1
2486413738368
i2
3
2486413738080
i3
2
2486413737664
addrs.setAtIndex(
ValueLayout.ADDRESS, 0, i1);
addrs.setAtIndex(
ValueLayout.ADDRESS, 1, i2);
addrs.setAtIndex(
ValueLayout.ADDRESS, 2, i3);
addrs
2486413738368
2486413738080
2486413737664
offset: 0
index: 0
offset: 8
index: 1
offset: 16
index: 2
Segment address: 2046322184240
limit: 24

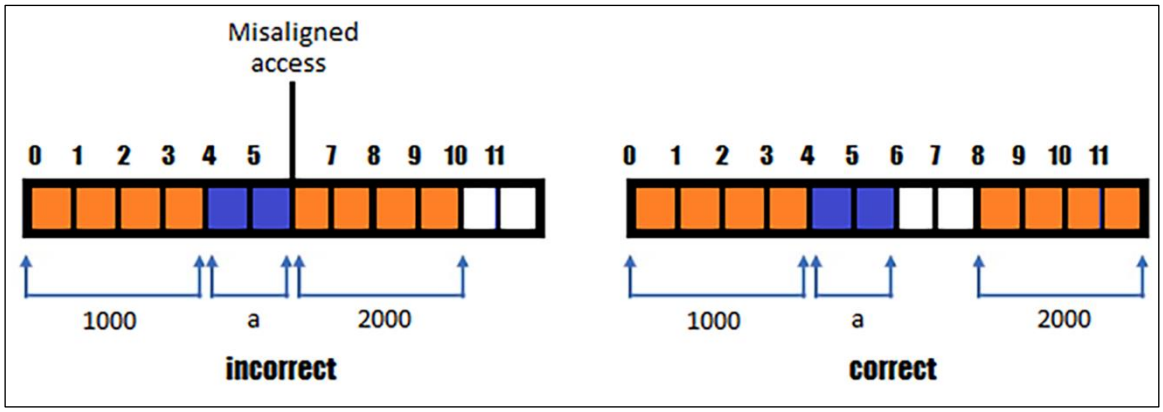

Segment address: 2046322184240
limit: 24
offset: 0
index: 0
offset: 8
index: 1
offset: 16
index: 2
2486413738368
2486413738080
2486413737664
addrs
MemorySegment addr1
= addrs.getAtIndex(
ValueLayout.ADDRESS, 0);
MemorySegment addr2
= addrs.getAtIndex(
ValueLayout.ADDRESS, 1);
MemorySegment addr3
= addrs.getAtIndex(
ValueLayout.ADDRESS,2)
address:
2486413738368
address:
2486413738080
address:
2486413737664
addr1
limit:0
addr2
limit:0
addr3
limit:0

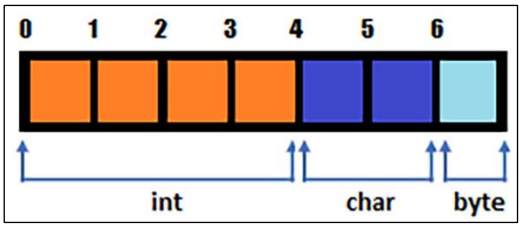

struct point {
double x;
double y;
};
struct point pointarr[5];
x
y
pointarr[0]
pointarr[0].x
pointarr[0].y
pointarr[1]
pointarr[1].x
pointarr[1].y
pointarr[4]
pointarr[4].x
pointarr[4].y
double = 8 bytes
double = 8 bytes
double x
double y

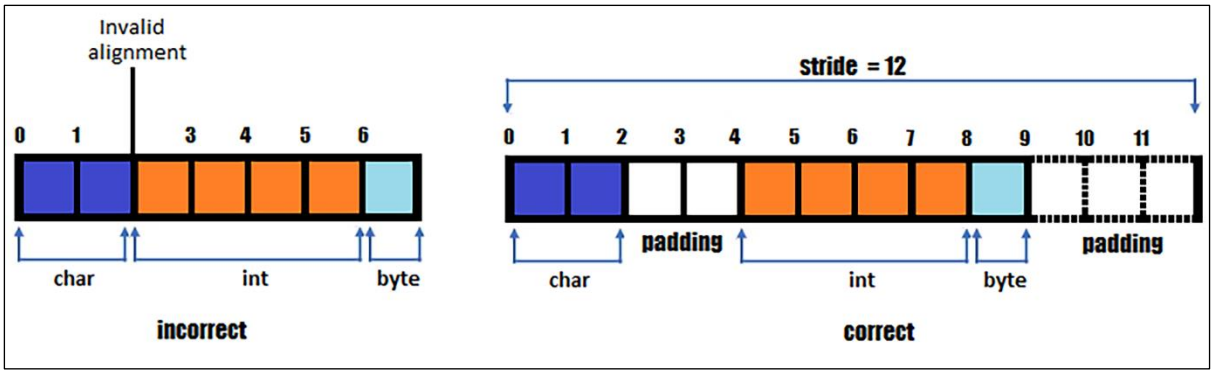

memory segment
of 80 bytes (160 bits)
double = 8 bytes
pair 1
pair 2
pair 3
pair 4
pair 5
x
y
x
y
x
y
x
y
x
y
index 0
index 1
index 2
index 3
index 4
index 5
index 6
index 7
index 8
index 9
offset 0
offset 16
offset 32
offset 48
offset 64
limit 80

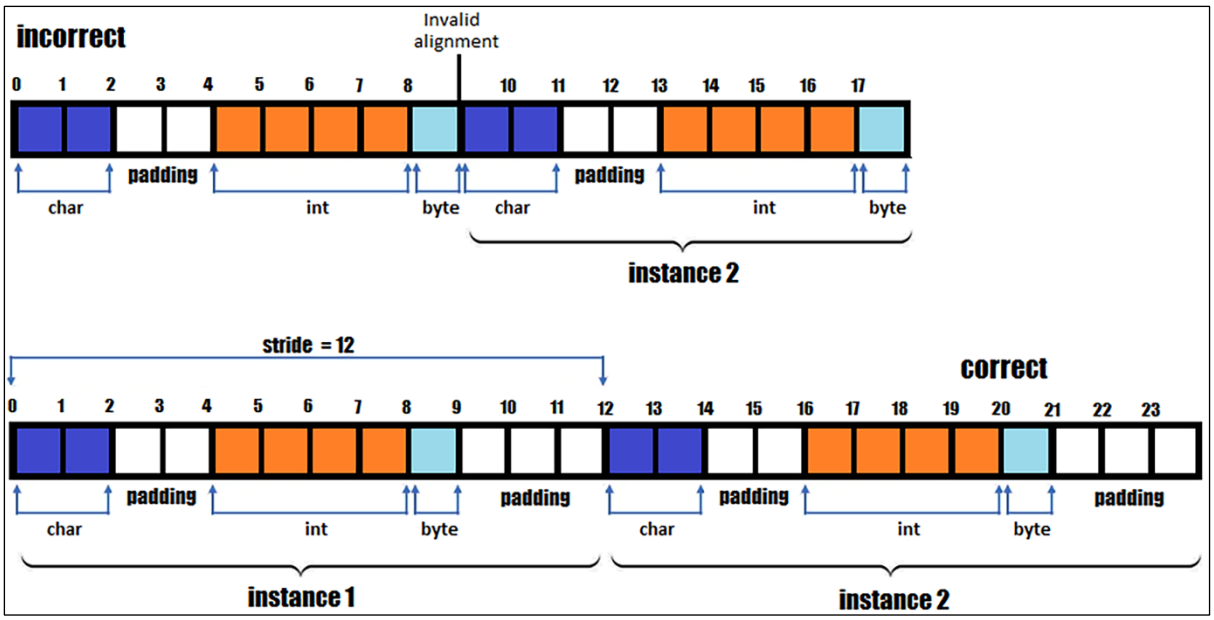

union product {
double price;
int sku;
} p;
price/sku
p
p.price/p.sku
double price
double = 8 bytes
int = 4 bytes
int sku

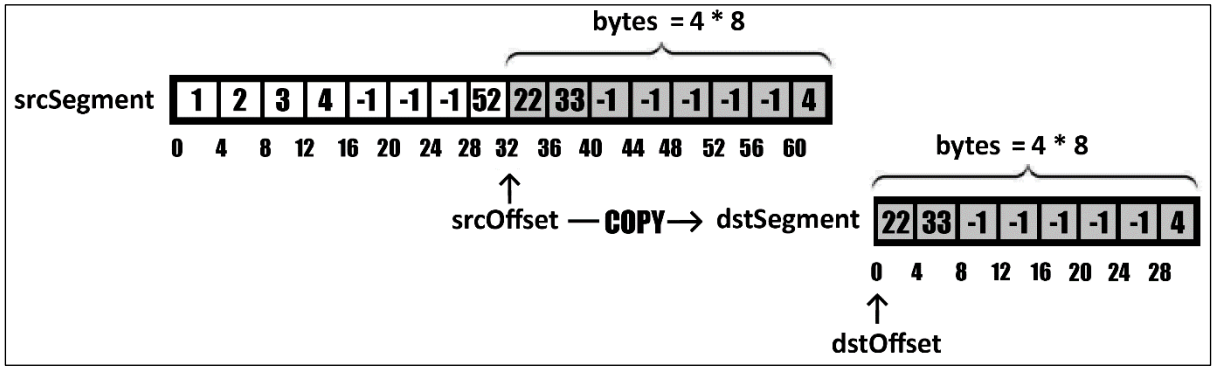

memory segment
8 bytes (16 bits)
0 1 2 3 4 5 6 7
offset 0
offset 4
limit 8
memory segment
16 bytes (128 bits)
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
padding
padding
offset 0
offset 4
offset 8
offset 12
limit 16

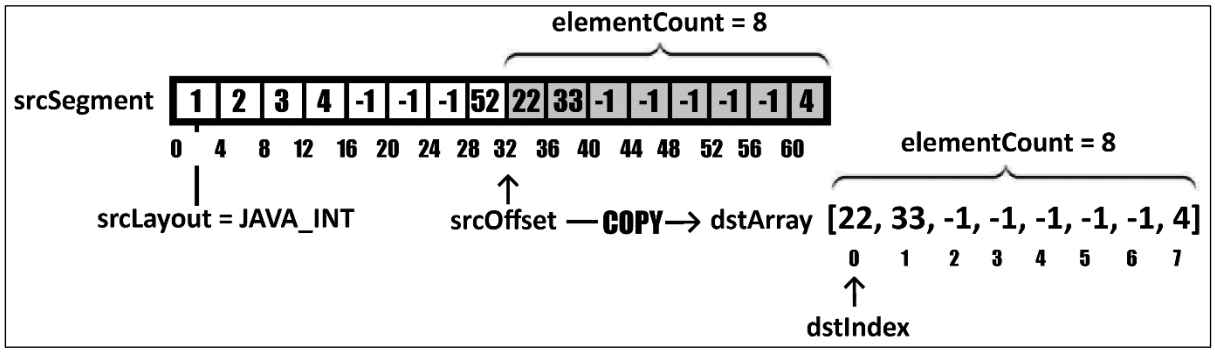

Misaligned
access
0 1 2 3 4 5 7 8 9 10 11
1000
a
2000
incorrect
0 1 2 3 4 5 6 7 8 9 10 11
1000
a
2000
correct

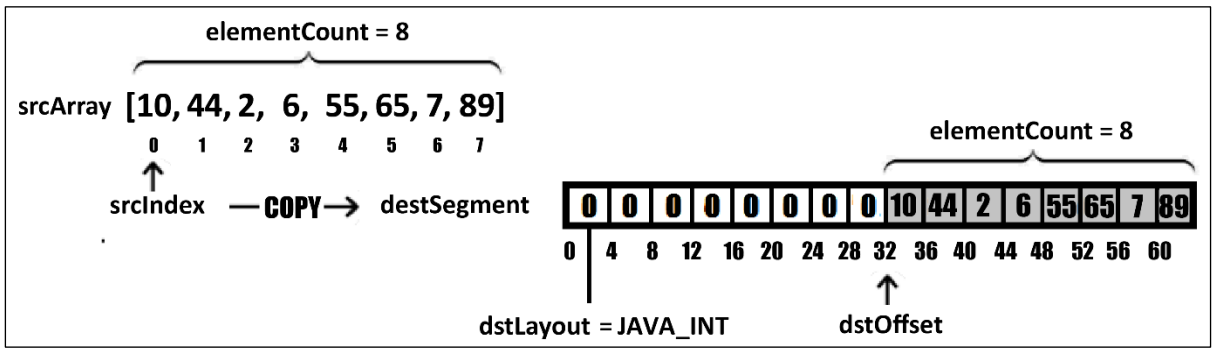

0 1 2 3 4 5 6
int
char
byte

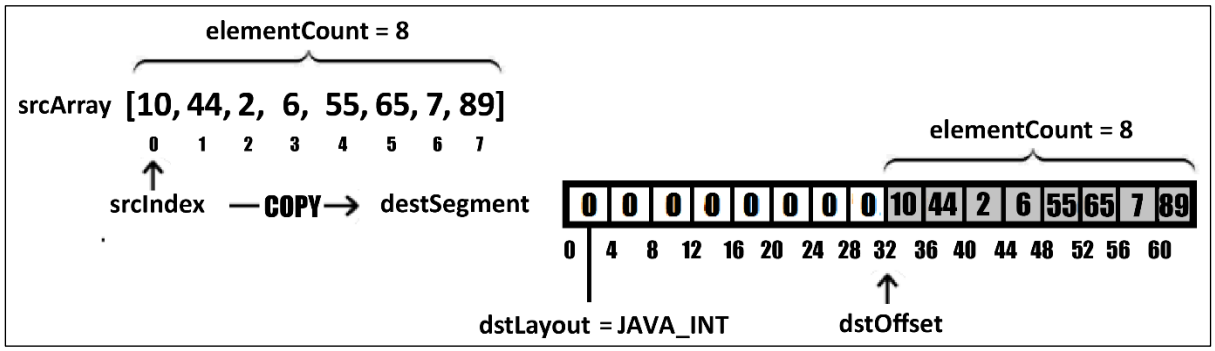

Invalid
alignment
0 1 3 4 5 6
char
int
byte
incorrect
stride = 12
0 1 2 3 4 5 6 7 8 9 10 11
padding
padding
char
int
byte
correct

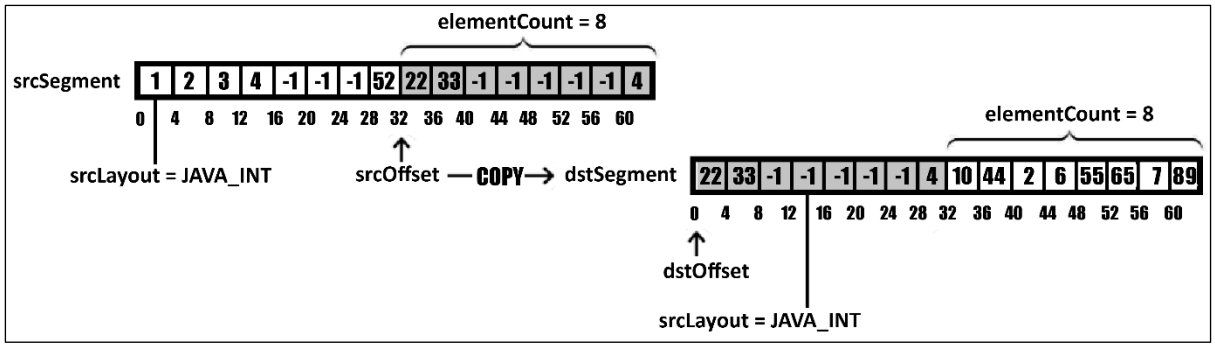

incorrect
Invalid
alignment
0 1 2 3 4 5 6 7 8 10 11 12 13 14 15 16 17
padding
padding
char
int
byte
char
int
byte
instance 2
stride = 12
correct
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
padding
padding
padding
padding
char
int
byte
char
int
byte
instance 1
instance 2

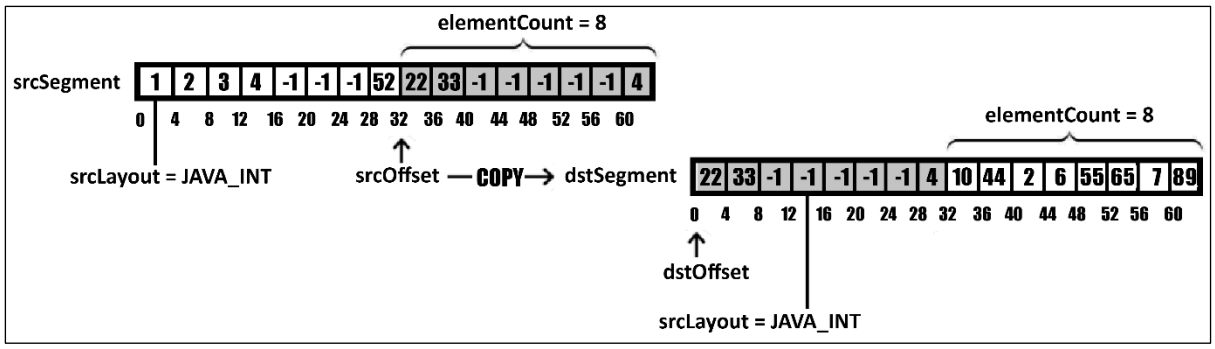

bytes = 4 * 8
srcSegment
1 2 3 4 -1 -1 -1 52 22 33 -1 -1 -1 -1 -1 4
0 4 8 12 16 20 24 28 32 36 40 44 48 52 56 60
srcOffset
COPY
dstSegment
bytes = 4 * 8
22 33 -1 -1 -1 -1 -1 4
0 4 8 12 16 20 24 28
dstOffset

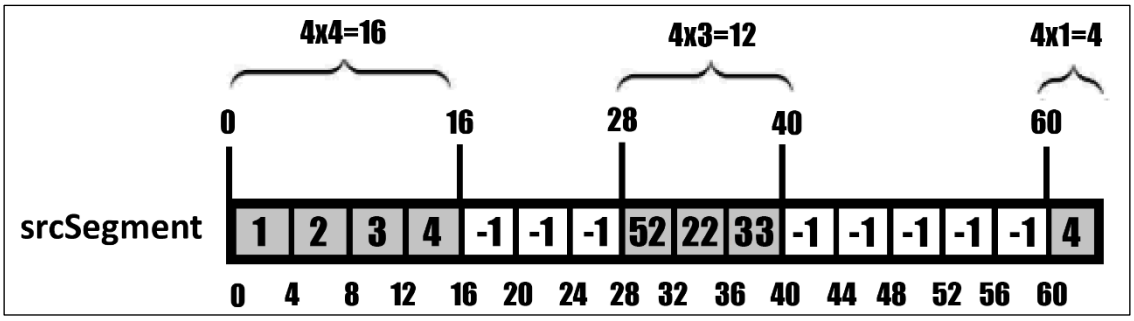

elementCount = 8
srcSegment
1 2 3 4 -1 -1 -1 52 22 33 -1 -1 -1 -1 -1 4
0 4 8 12 16 20 24 28 32 36 40 44 48 52 56 60
srcLayout = JAVA_INT
srcOffset
COPY
dstArray
elementCount = 8
[22, 33, -1, -1, -1, -1, -1, 4]
0 1 2 3 4 5 6 7
dstIndex

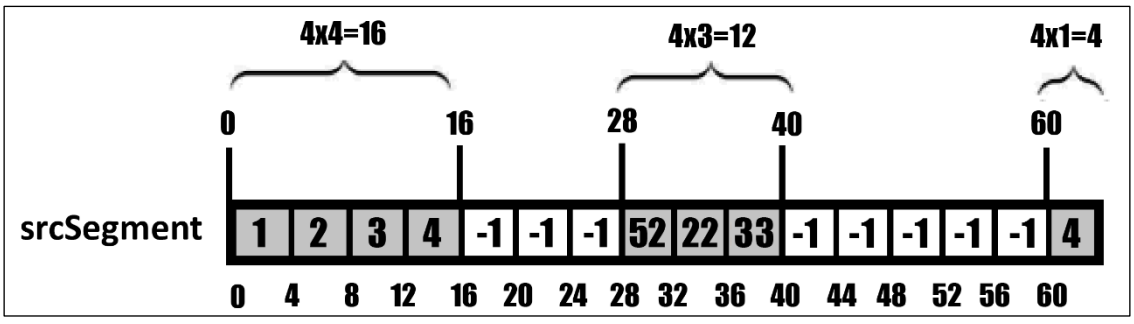

elementCount = 8
srcArray
[10, 44, 2, 6, 55, 65, 7, 89]
0 1 2 3 4 5 6 7
srcIndex
COPY
destSegment
elementCount = 8
0 0 0 0 0 0 0 0 10 44 2 6 55 65 7 89
0 4 8 12 16 20 24 28 32 36 40 44 48 52 56 60
dstLayout = JAVA_INT
dstOffset


elementCount = 8
srcArray
[10, 44, 2, 6, 55, 65, 7, 89]
0 1 2 3 4 5 6 7
srcIndex
COPY
destSegment
elementCount = 8
0 0 0 0 0 0 0 0 10 44 2 6 55 65 7 89
0 4 8 12 16 20 24 28 32 36 40 44 48 52 56 60
dstLayout = JAVA_INT
dstOffset



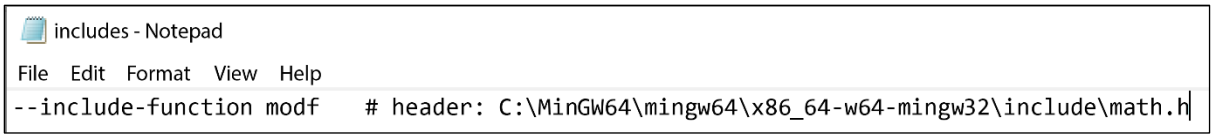


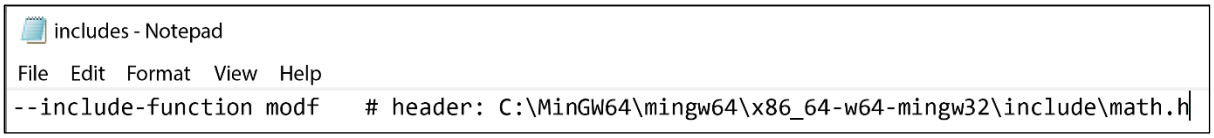





```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter07\P171_JextractAndModf>
  jextract  --source --output src\main\java -t c.lib.math
 -I C:\MinGW64\mingw64\x86_64-w64-mingw32\include
  C:\MinGW64\mingw64\x86_64-w64-mingw32\include\math.h
```

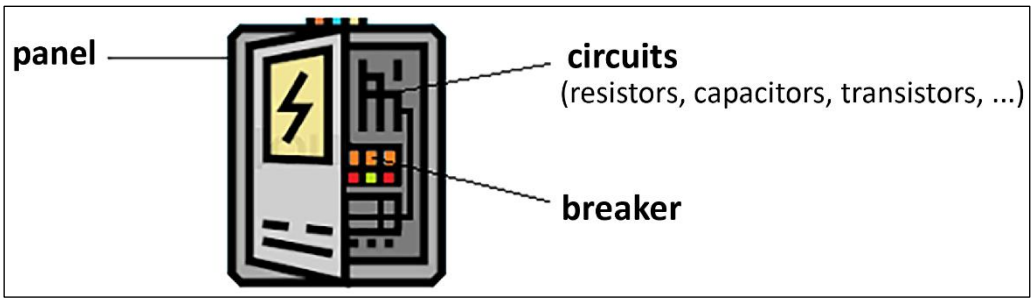
```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter07\P171_JextractAndModf>
  jextract --dump-includes includes.txt
  -I C:\MinGW64\mingw64\x86_64-w64-mingw32\include
  C:\MinGW64\mingw64\x86_64-w64-mingw32\include\math.h
```

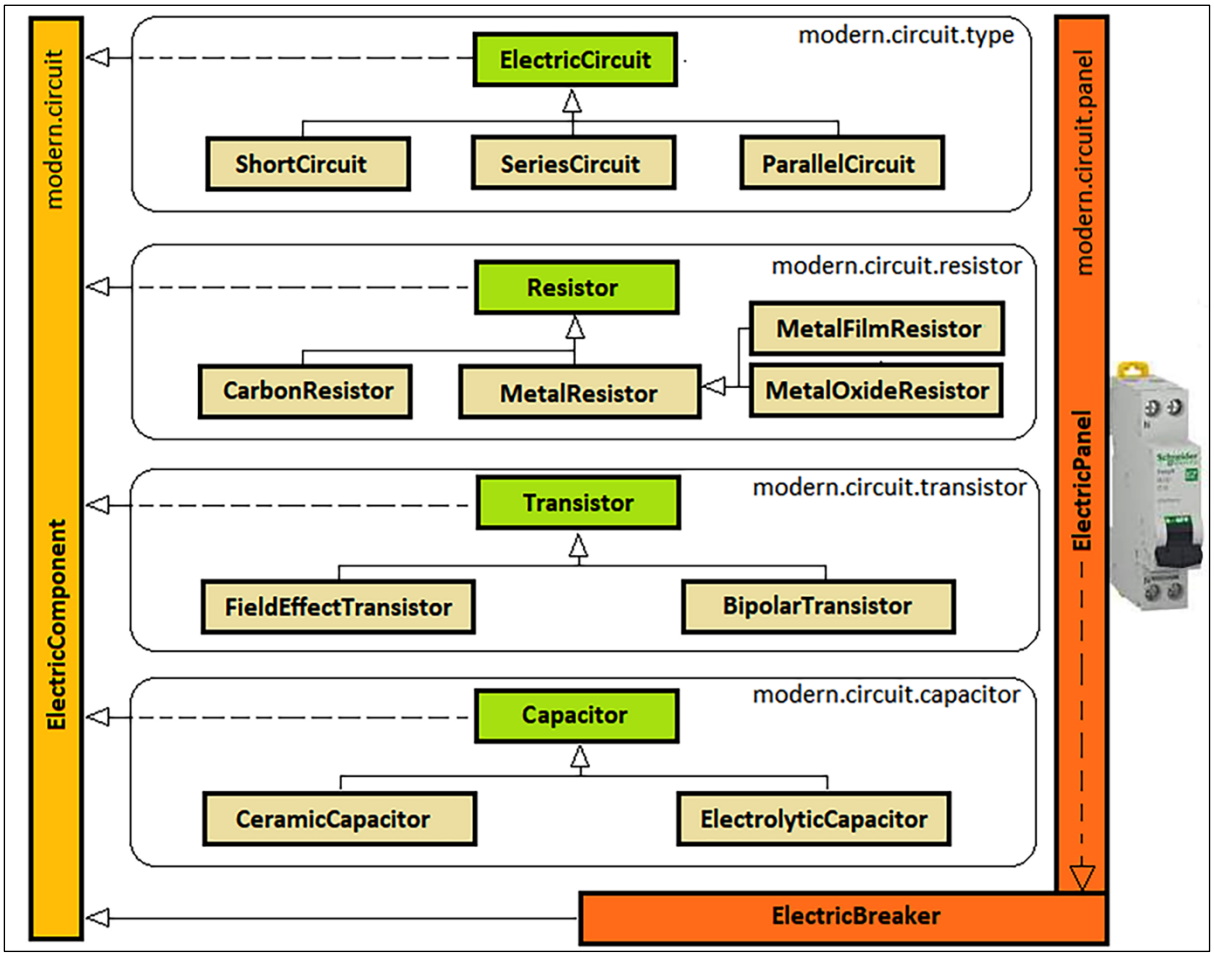




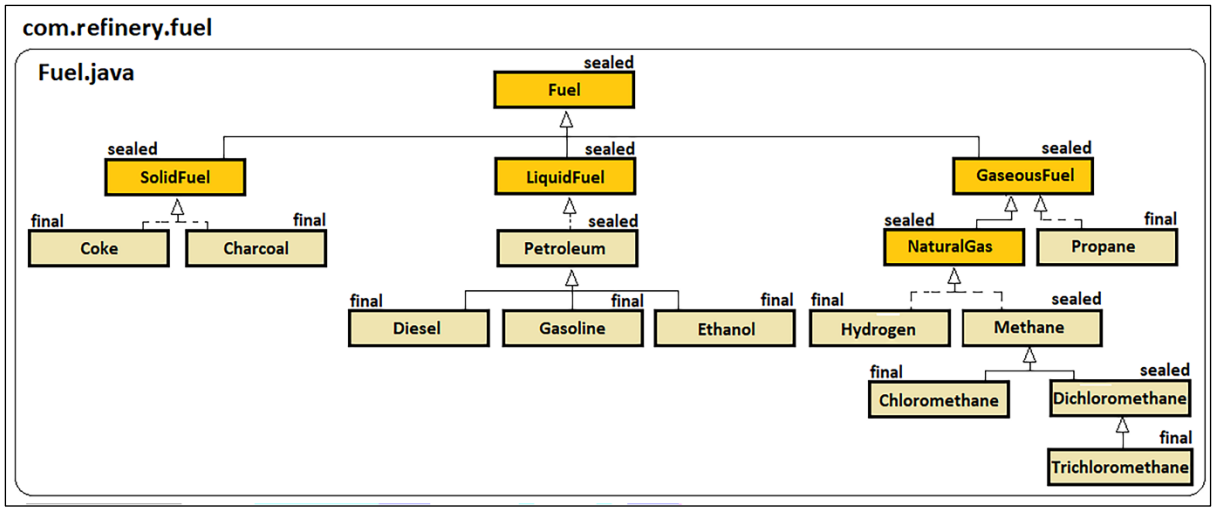

```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter07\P171_JextractAndModf>
  jextract --source @includes.txt --output src\main\java -t c.lib.math
  -I C:\MinGW64\mingw64\x86_64-w64-mingw32\include
  C:\MinGW64\mingw64\x86_64-w64-mingw32\include\math.h
```

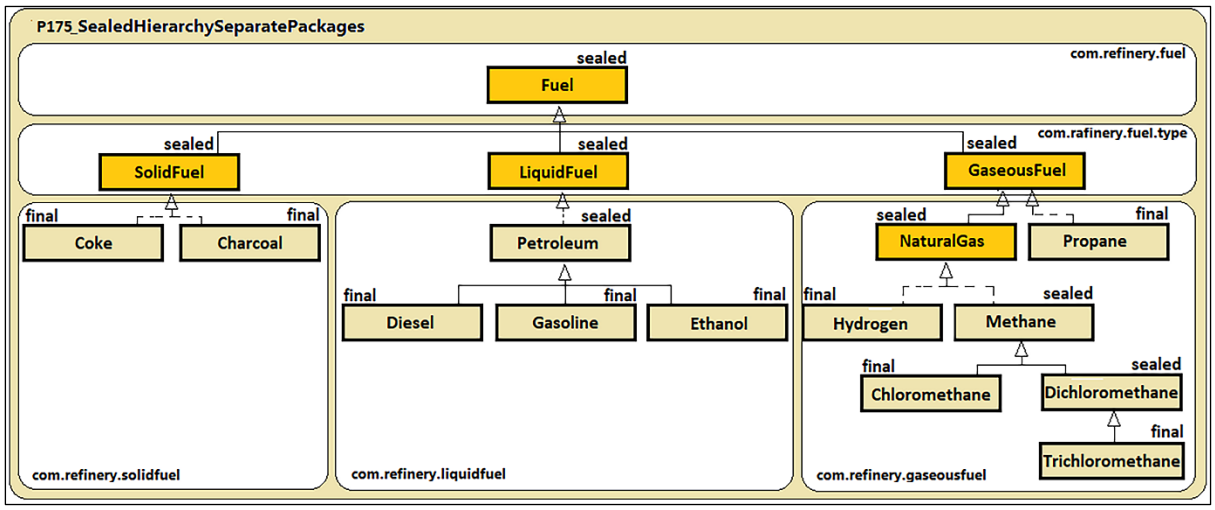
```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter07\P171_JextractAndModf>
  jextract @includes.txt --output target\classes -t c.lib.math
  -I C:\MinGW64\mingw64\x86_64-w64-mingw32\include
  C:\MinGW64\mingw64\x86_64-w64-mingw32\include\math.h
```

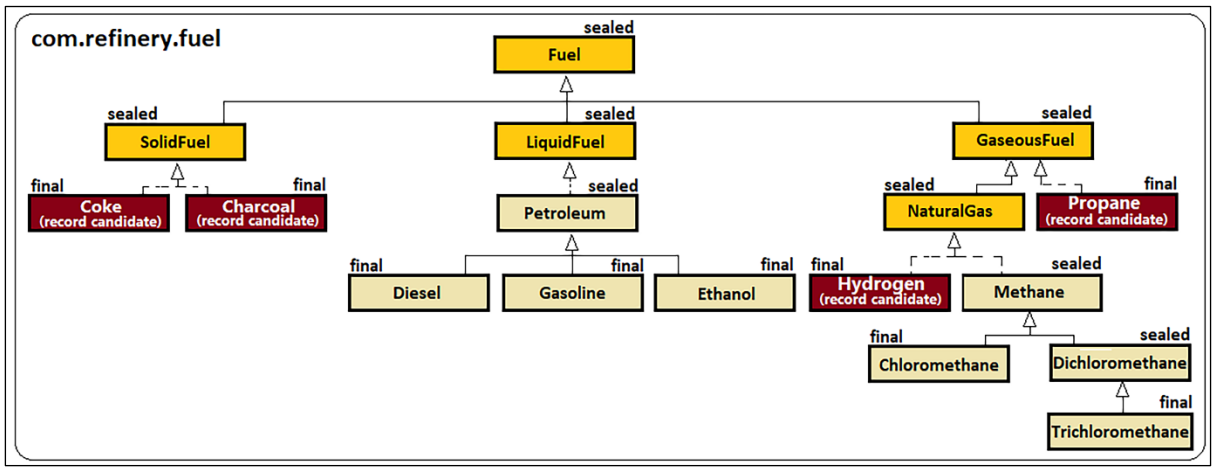

panel
circuits
(resistors, capacitors, transistors, ...)
breaker

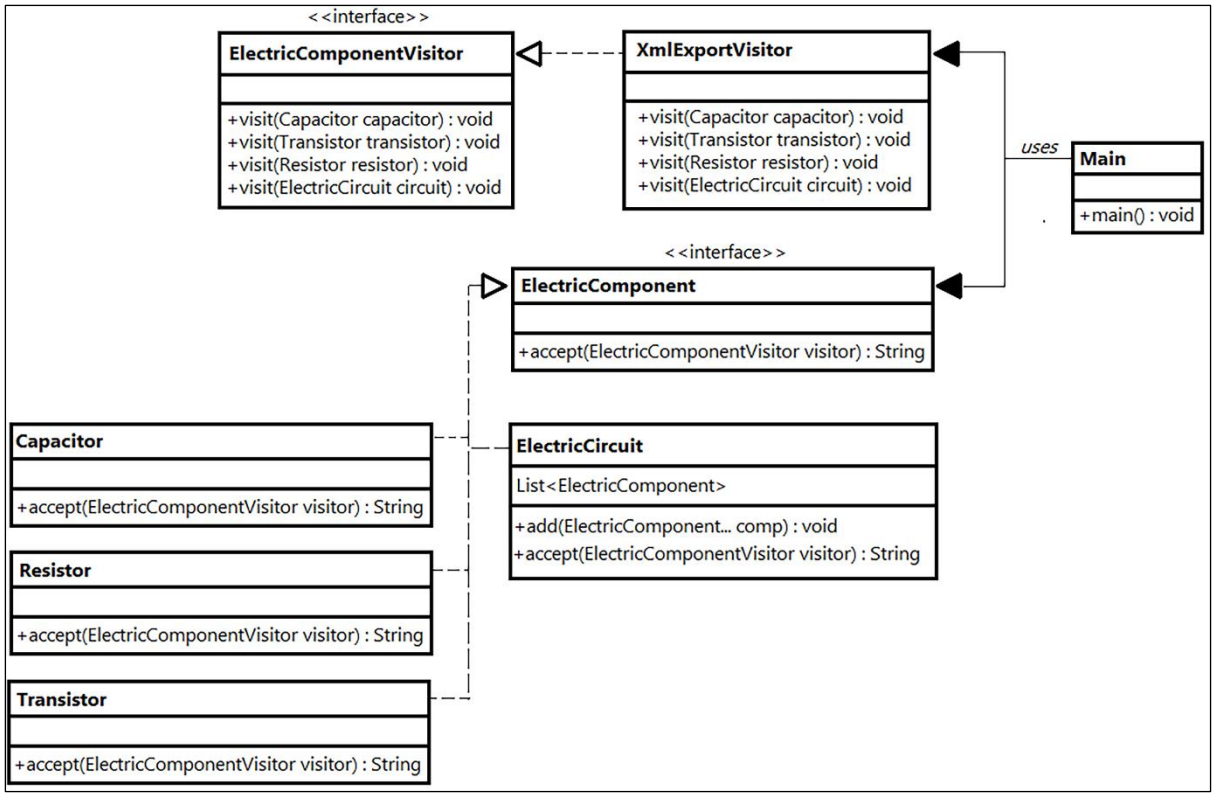

modern.circuit
modern.circuit.type
ElectricCircuit
ShortCircuit
SeriesCircuit
ParallelCircuit
modern.circuit.resistor
Resistor
MetalFilmResistor
CarbonResistor
MetalResistor
MetalOxideResistor
modern.circuit.transistor
Transistor
FieldEffectTransistor
BipolarTransistor
modern.circuit.capacitor
Capacitor
CeramicCapacitor
ElectrolyticCapacitor
ElectricComponent
modern.circuit.panel
ElectricPanel
ElectricBreaker

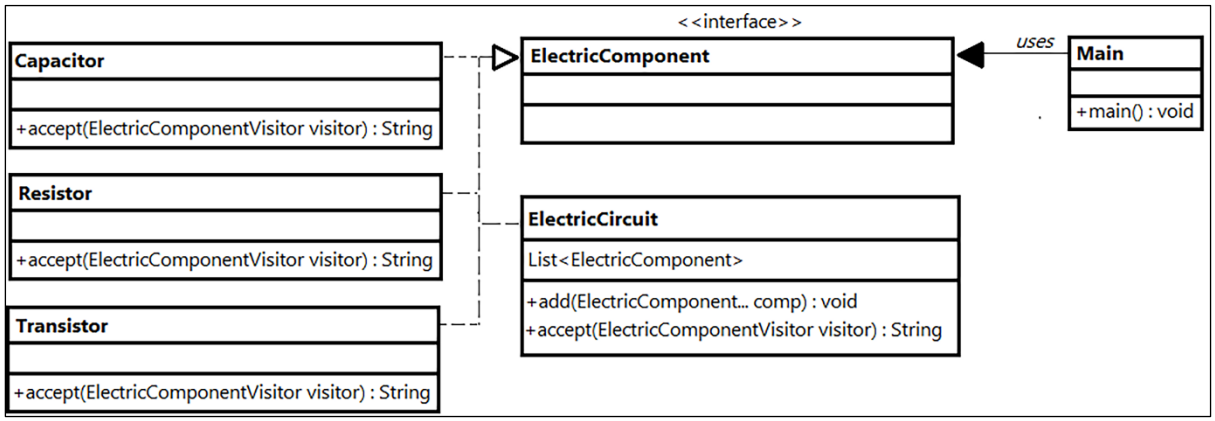
final
JDK 17
Sealed
Classes
JEP 409
package
-private


com.refinery.fuel
Fuel.java
sealed
Fuel
sealed
SolidFuel
sealed
LiquidFuel
sealed
GaseousFuel
final
Coke
final
Charcoal
sealed
Petroleum
sealed
NaturalGas
final
Propane
final
Diesel
final
Gasoline
final
Ethanol
final
Hydrogen
sealed
Methane
final
Chloromethane
sealed
Dichloromethane
final
Trichloromethane

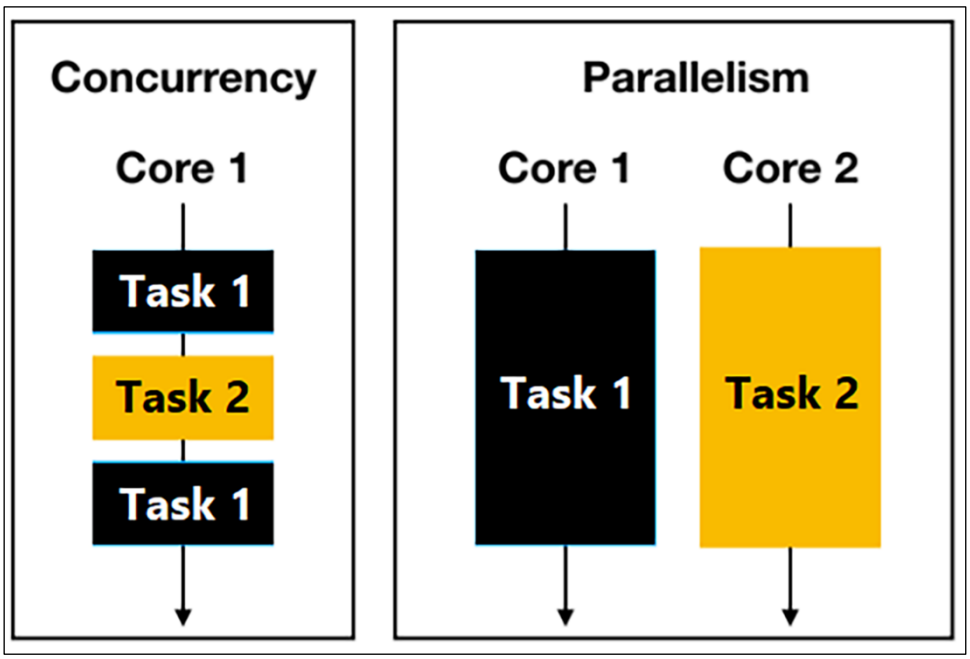

P175_SealedHierarchySeparatePackages
com.refinery.fuel
sealed
Fuel
com.rafinery.fuel.type
sealed
SolidFuel
sealed
LiquidFuel
sealed
GaseousFuel
final
Coke
final
Charcoal
sealed
Petroleum
final
Diesel
final
Gasoline
final
Ethanol
sealed
NaturalGas
final
Propane
final
Hydrogen
sealed
Methane
final
Chloromethane
sealed
Dichloromethane
final
Trichloromethane
com.refinery.solidfuel
com.refinery.liquidfuel
com.refinery.gaseousfuel

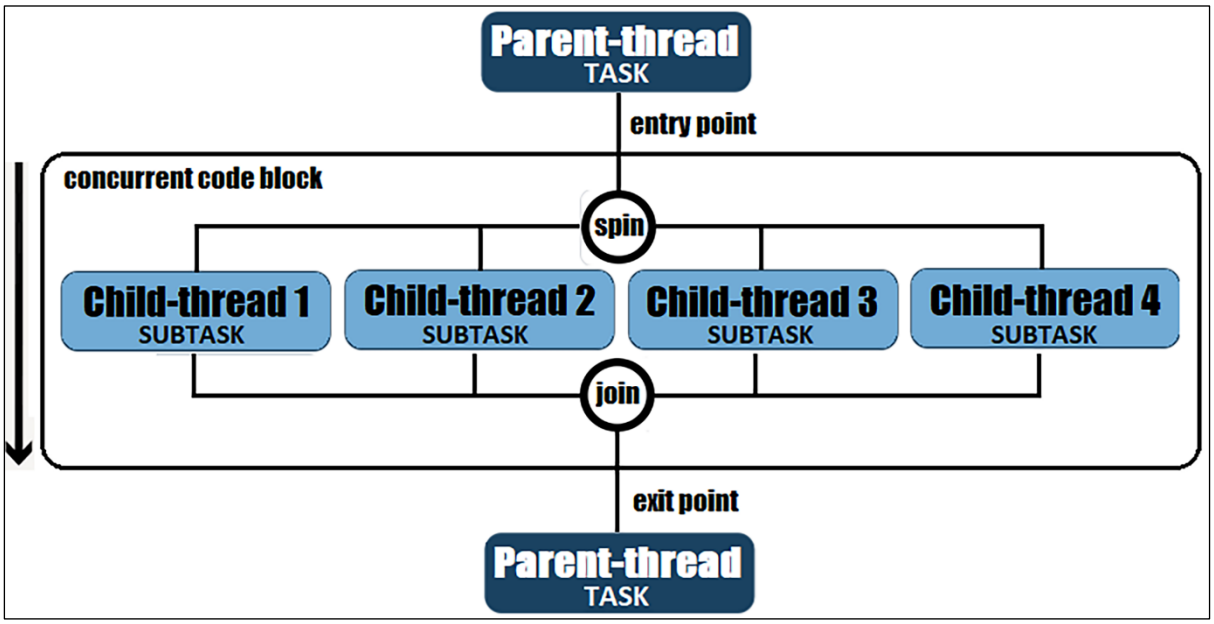

com.refinery.fuel
sealed
Fuel
sealed
SolidFuel
sealed
LiquidFuel
sealed
GaseousFuel
final
Coke
(record candidate)
final
Charcoal
(record candidate)
sealed
Petroleum
sealed
NaturalGas
final
Propane
(record candidate)
final
Diesel
final
Gasoline
final
Ethanol
final
Hydrogen
(record candidate)
sealed
Methane
final
Chloromethane
sealed
Dichloromethane
final
Trichloromethane


<<interface>>
ElectricComponentVisitor
+visit(Capacitor capacitor) : void
+visit(Transistor transistor) : void
+visit(Resistor resistor) : void
+visit(ElectricCircuit circuit) : void
XmlExportVisitor
+visit(Capacitor capacitor) : void
+visit(Transistor transistor) : void
+visit(Resistor resistor) : void
+visit(ElectricCircuit circuit) : void
uses
Main
+main() : void
<<interface>>
ElectricComponent
+accept(ElectricComponentVisitor visitor) : String
Capacitor
+accept(ElectricComponentVisitor visitor) : String
ElectricCircuit
List<ElectricComponent>
+add(ElectricComponent... comp) : void
+accept(ElectricComponentVisitor visitor) : String
Resistor
+accept(ElectricComponentVisitor visitor) : String
Transistor
+accept(ElectricComponentVisitor visitor) : String

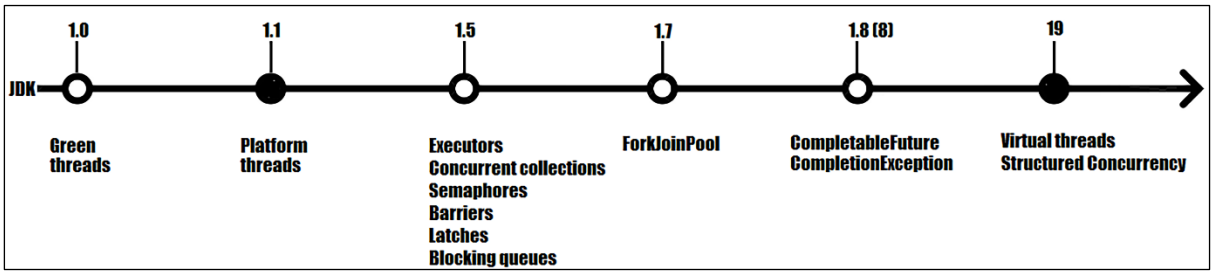

<<interface>>
Capacitor
+accept(ElectricComponentVisitor visitor) : String
ElectricComponent
uses
Main
+main() : void
Resistor
+accept(ElectricComponentVisitor visitor) : String
ElectricCircuit
List<ElectricComponent>
+add(ElectricComponent... comp) : void
+accept(ElectricComponentVisitor visitor) : String
Transistor
+accept(ElectricComponentVisitor visitor) : String

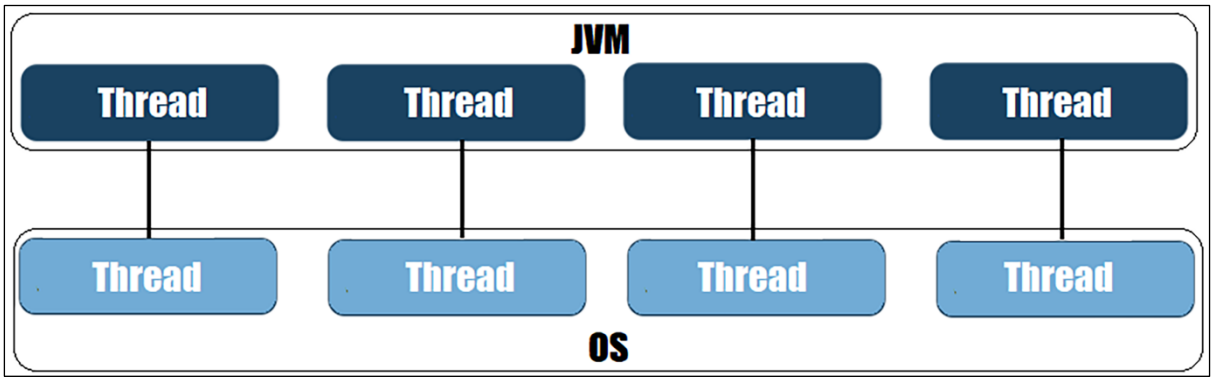
Chapter 10: Concurrency - Virtual Threads and Structured Concurrency

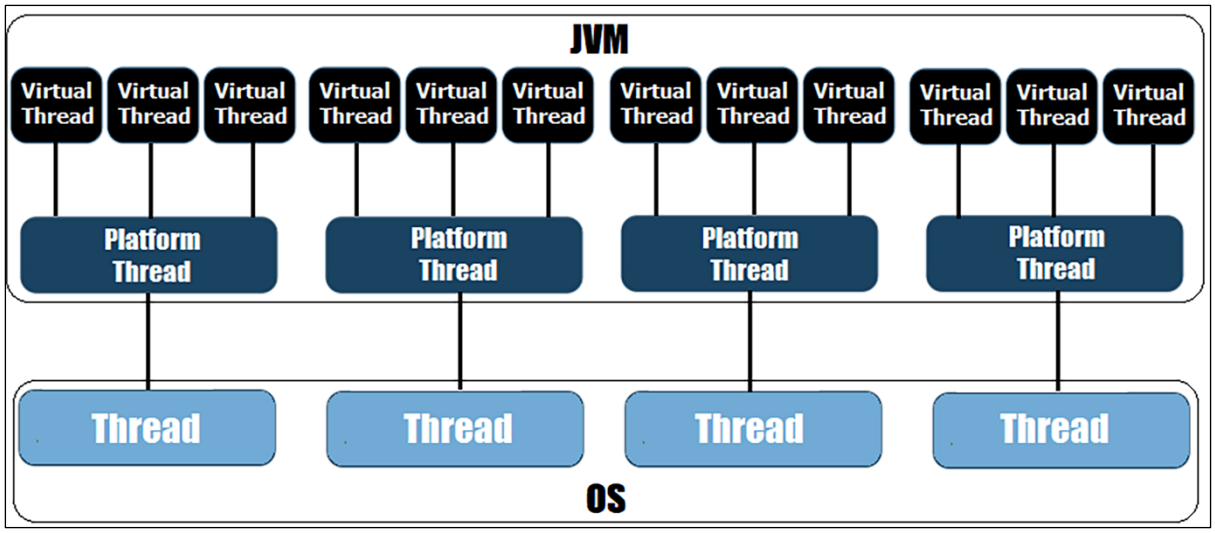


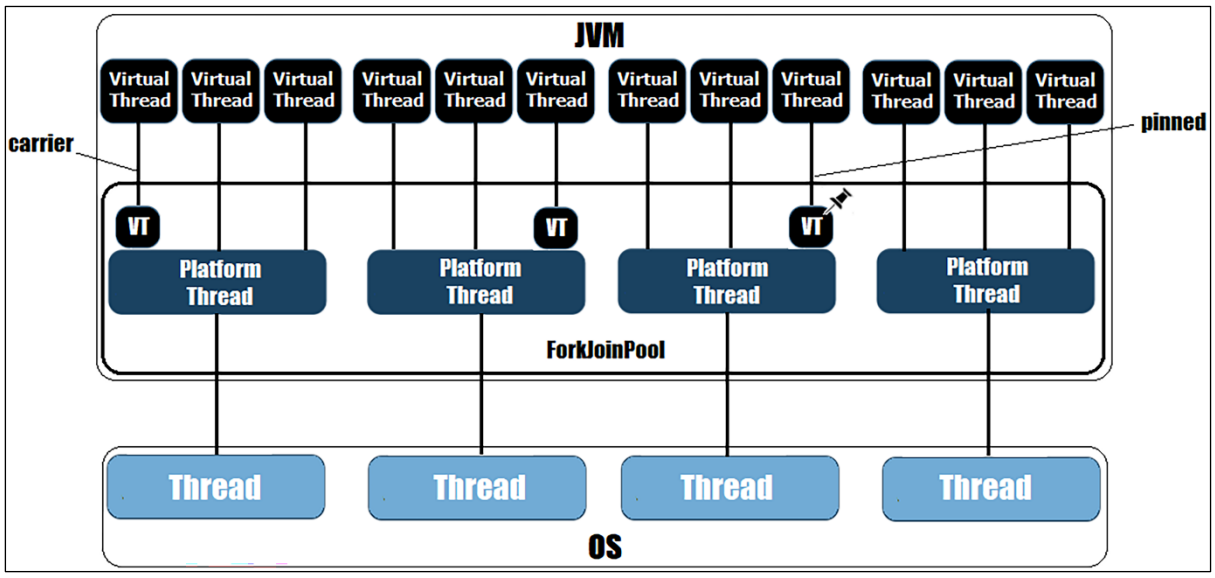


DON'T
REUSE
VIRTUAL
THREADS


JDK
1.0
1.1
1.5
1.7
1.8 (8)
19
Green threads
Platform threads
Executors
Concurrent collections
Semaphores
Barriers
Latches
Blocking queues
ForkJoinPool
CompletableFuture
CompletionException
Virtual threads
Structured Concurrency


JVM
Thread
Thread
Thread
Thread
Thread
Thread
Thread
Thread
OS






```
[10:46:51]  5 Thread[#27,Thread-5,5,main]  |  [10:46:54]  7 Thread[#29,Thread-7,5,main]
[10:46:51]  1 Thread[#23,Thread-1,5,main]  |  [10:46:54]  0 Thread[#22,Thread-0,5,main]
[10:46:51]  9 Thread[#31,Thread-9,5,main]  |  [10:46:54]  8 Thread[#30,Thread-8,5,main]
[10:46:51]  4 Thread[#26,Thread-4,5,main]  |  [10:46:54]  6 Thread[#28,Thread-6,5,main]
[10:46:51]  7 Thread[#29,Thread-7,5,main]  |  [10:46:54]  4 Thread[#26,Thread-4,5,main]
[10:46:51]  0 Thread[#22,Thread-0,5,main]  |  [10:46:54]  9 Thread[#31,Thread-9,5,main]
[10:46:51]  6 Thread[#28,Thread-6,5,main]  |  [10:46:54]  1 Thread[#23,Thread-1,5,main]
[10:46:51]  8 Thread[#30,Thread-8,5,main]  |  [10:46:54]  5 Thread[#27,Thread-5,5,main]
[10:46:51]  3 Thread[#25,Thread-3,5,main]  |  [10:46:54]  3 Thread[#25,Thread-3,5,main]
[10:46:51]  2 Thread[#24,Thread-2,5,main]  |  [10:46:54]  2 Thread[#24,Thread-2,5,main]
```

first logging | sleep | second logging

```
[07:46:50]  4  [#27]/runnable@ForkJoinPool-1-worker-5
[07:46:50]  3  [#26]/runnable@ForkJoinPool-1-worker-4
[07:46:50]  1  [#24]/runnable@ForkJoinPool-1-worker-2
[07:46:50]  2  [#25]/runnable@ForkJoinPool-1-worker-3
[07:46:50]  0  [#22]/runnable@ForkJoinPool-1-worker-1
[07:46:50]  7  [#30]/runnable@ForkJoinPool-1-worker-8
[07:46:50]  9  [#32]/runnable@ForkJoinPool-1-worker-7
[07:46:50]  8  [#31]/runnable@ForkJoinPool-1-worker-7
[07:46:50]  6  [#29]/runnable@ForkJoinPool-1-worker-7
[07:46:50]  5  [#28]/runnable@ForkJoinPool-1-worker-6
```

first logging

```
[07:46:53]  1  [#24]/runnable@ForkJoinPool-1-worker-1
[07:46:53]  2  [#25]/runnable@ForkJoinPool-1-worker-4
[07:46:53]  9  [#32]/runnable@ForkJoinPool-1-worker-7
[07:46:53]  7  [#30]/runnable@ForkJoinPool-1-worker-5
[07:46:53]  4  [#27]/runnable@ForkJoinPool-1-worker-8
[07:46:53]  0  [#22]/runnable@ForkJoinPool-1-worker-2
[07:46:53]  8  [#31]/runnable@ForkJoinPool-1-worker-3
[07:46:53]  3  [#26]/runnable@ForkJoinPool-1-worker-6
[07:46:53]  5  [#28]/runnable@ForkJoinPool-1-worker-4
[07:46:53]  6  [#29]/runnable@ForkJoinPool-1-worker-1
```

sleep **second logging**

Administrator: Command Prompt

```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter10\P224_MonitoringVirtualThreadsJfr1 >
java --enable-preview -XX:StartFlightRecording=filename=recording.jfr,settings=vtEvent.jfc
     -classpath target/classes modern.challenge.Main
```

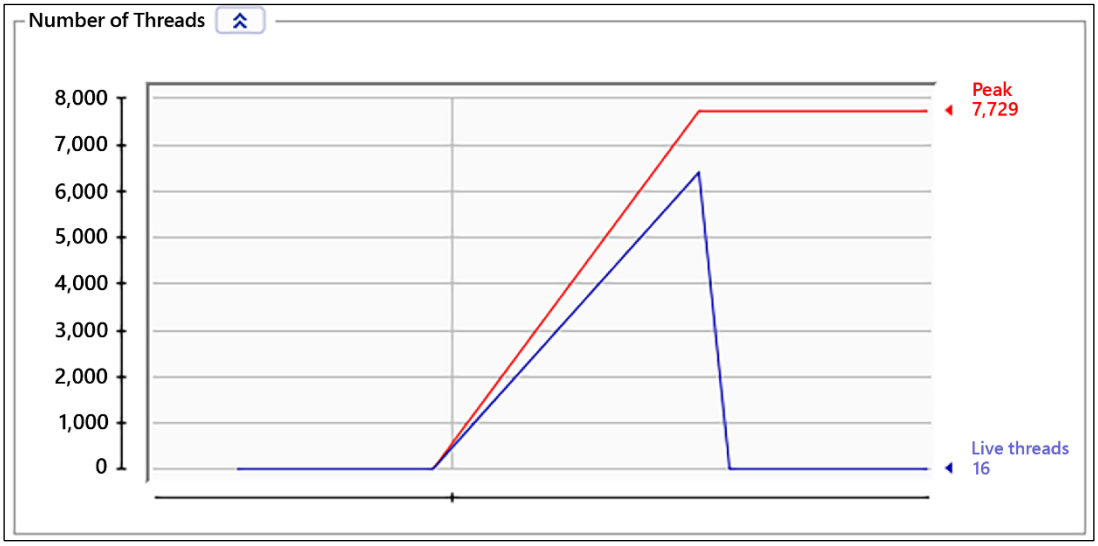
Administrator: Command Prompt

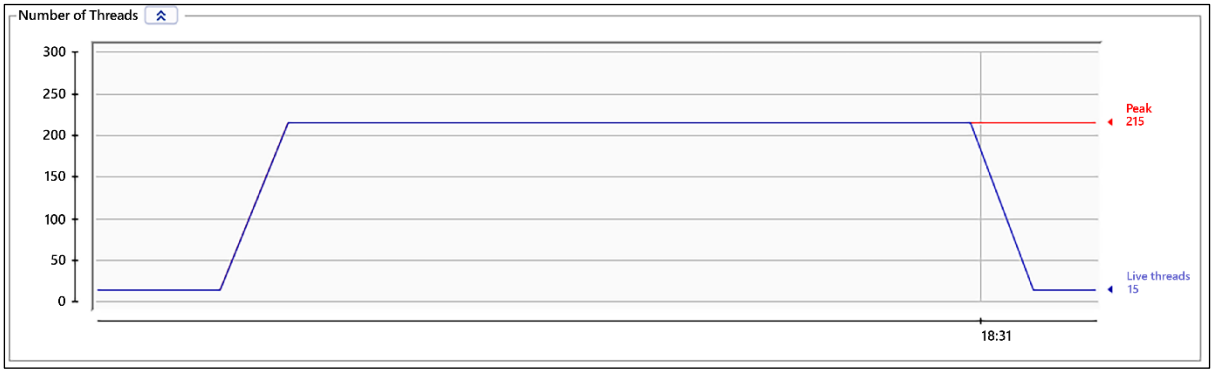
```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter10\P224_MonitoringVirtualThreadsJfr1>
  jfr print recording.jfr

jdk.VirtualThreadStart {
  startTime = 15:55:21.215 (2023-06-03)
  javaThreadId = 25
  eventThread = "" (javaThreadId = 25, virtual)
  stackTrace = [
    java.lang.VirtualThread.run(Runnable) line: 300
    java.lang.VirtualThread$VThreadContinuation.lambda$new$0(VirtualThread, Runnable) line: 177
    jdk.internal.vm.Continuation.enter0() line: 327
    jdk.internal.vm.Continuation.enter(Continuation, boolean) line: 320
    jdk.internal.vm.Continuation.enterSpecial(Continuation, boolean, boolean)
  ]
}
```

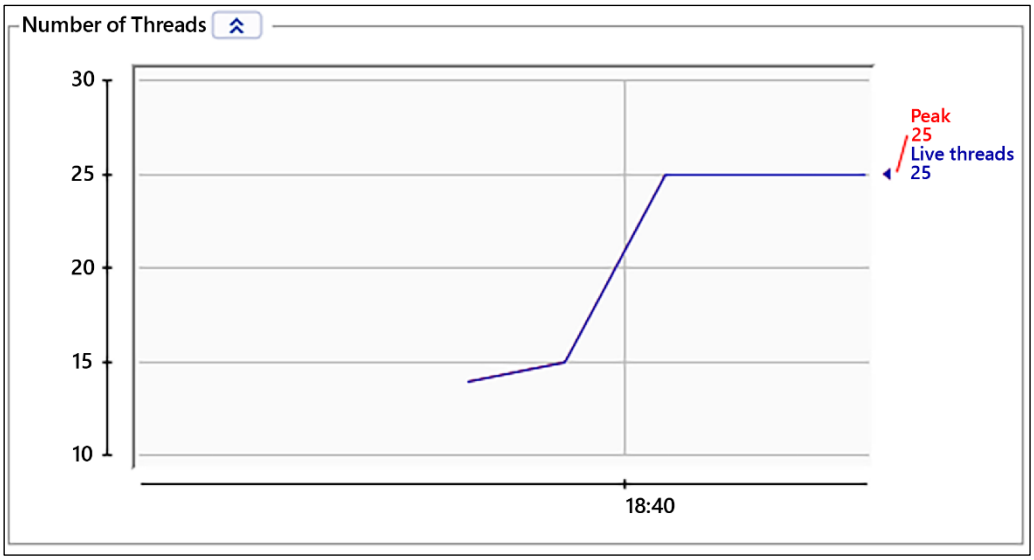
Administrator: Command Prompt

```
C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter10\P224_MonitoringVirtualThreadsJfr1 >
  jfr print recording.jfr

jdk.VirtualThreadEnd {
  startTime = 15:55:57.864 (2023-06-03)
  javaThreadId = 25
  eventThread = "" (javaThreadId = 25, virtual)
}
```

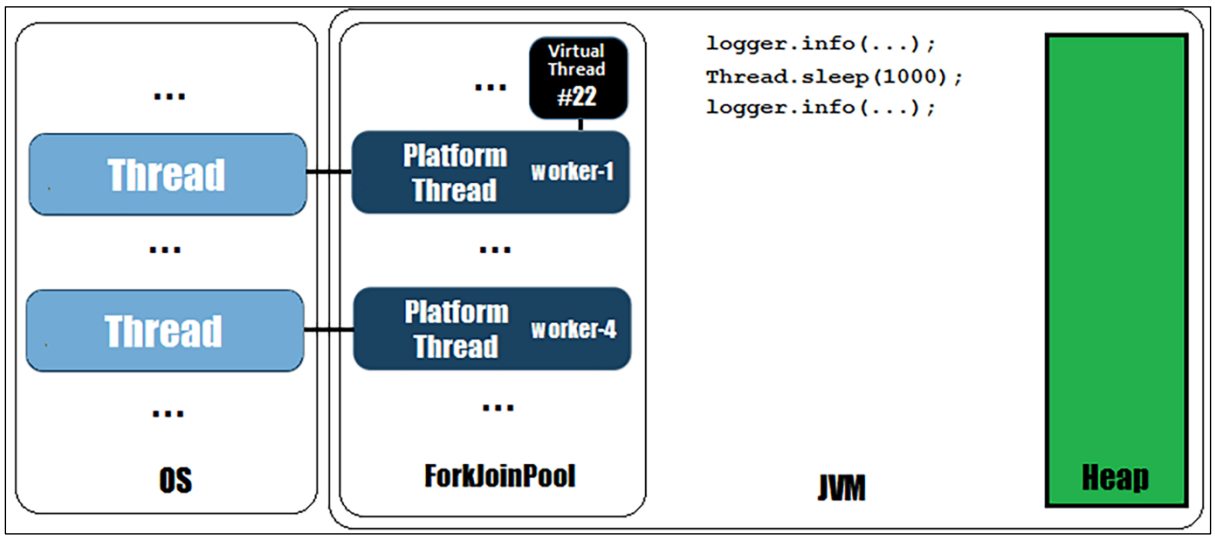
```
jdk.VirtualThreadPinned {
  startTime = 08:01:42.424 (2023-06-04)
  duration = 1.97 s
  eventThread = "" (javaThreadId = 28, virtual)
  stackTrace = [
    java.lang.VirtualThread.parkOnCarrierThread(boolean, long) line: 645
    java.lang.VirtualThread.parkNanos(long) line: 611
    java.lang.VirtualThread.doSleepNanos(long) line: 777
    java.lang.VirtualThread.sleepNanos(long) line: 750
    java.lang.Thread.sleep(Duration) line: 557
    ...
  ]
}
```

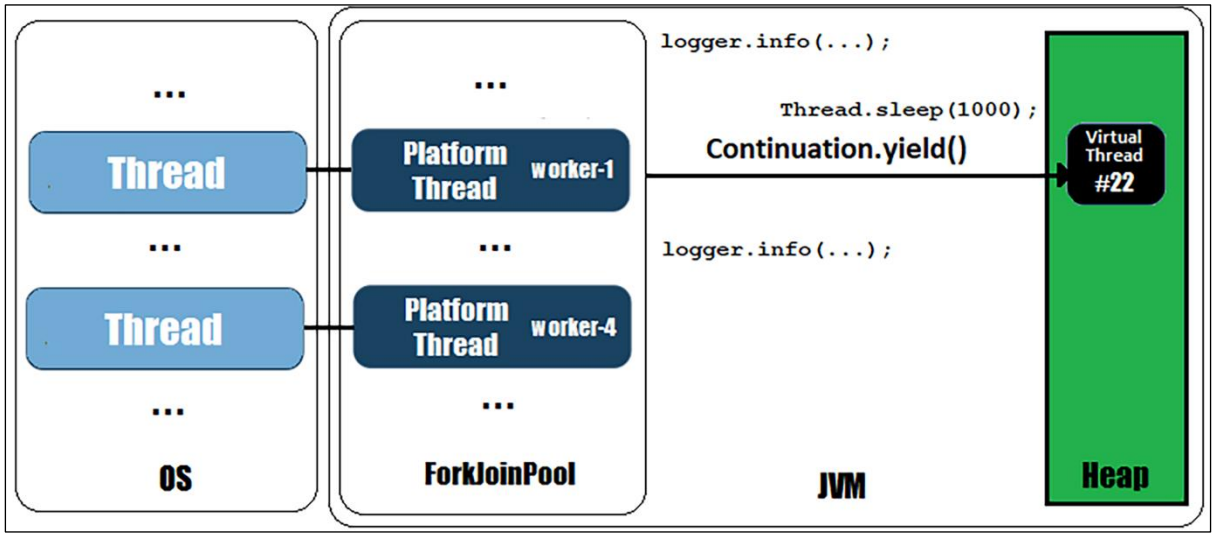


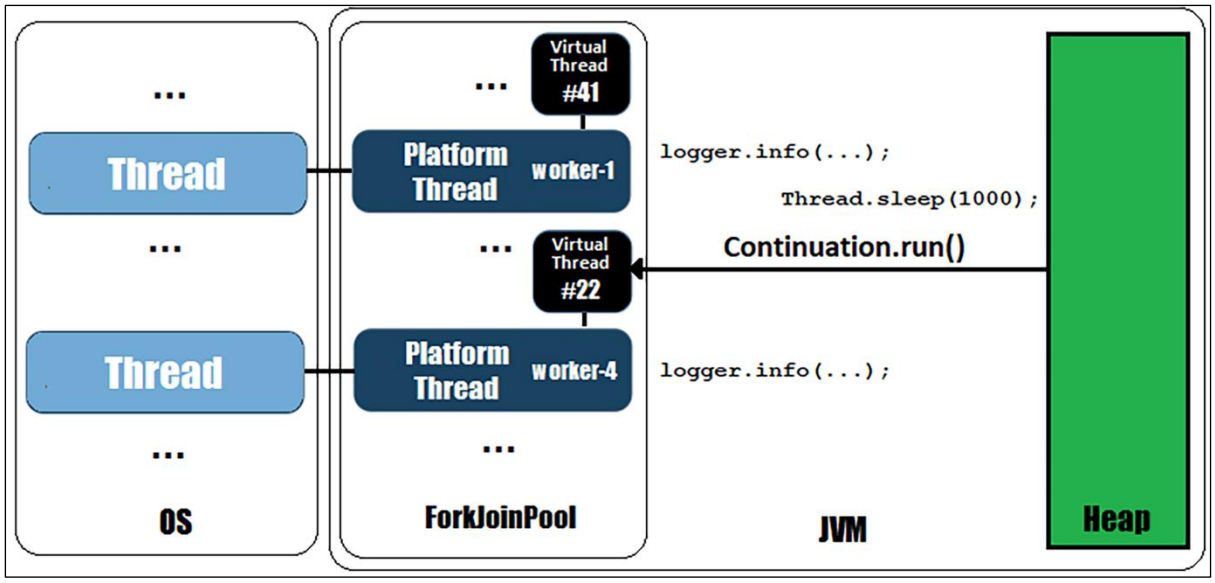


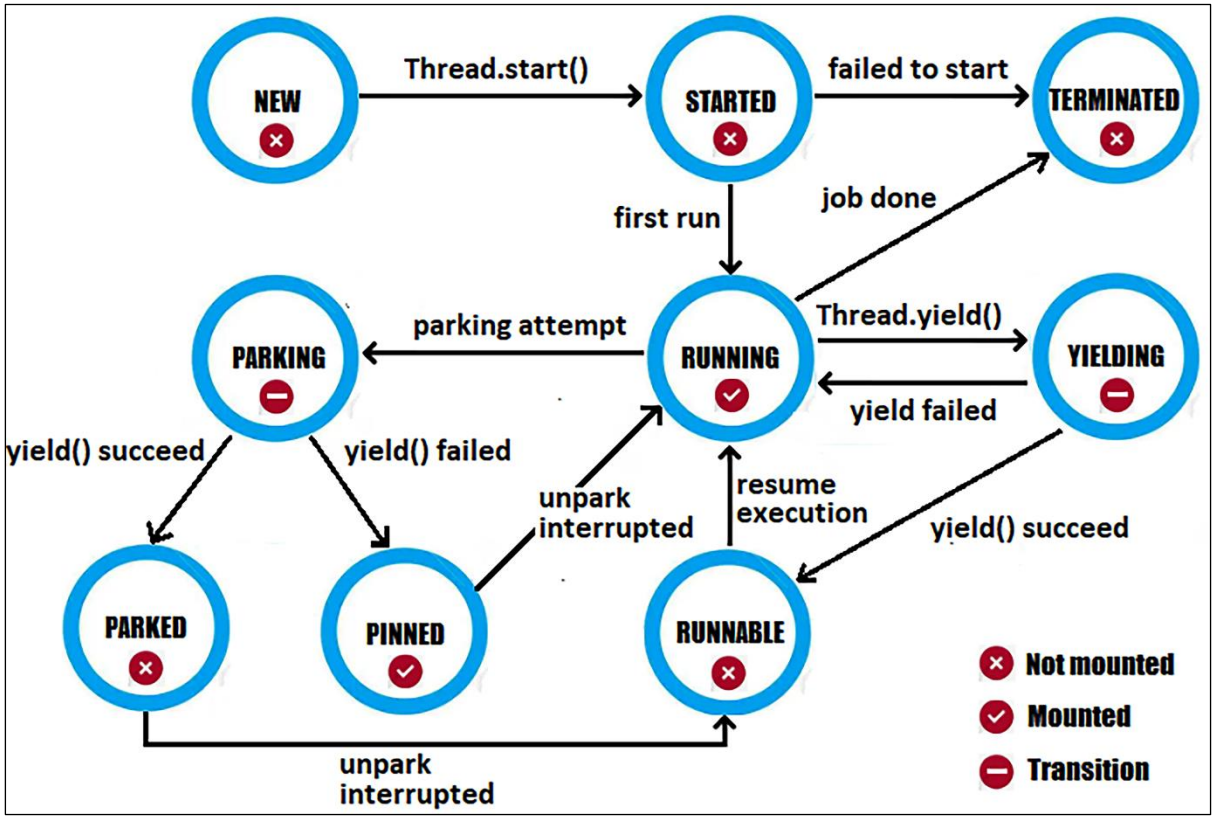

Number of Threads
30
25
20
15
10
18:40
Peak
25
Live threads
25

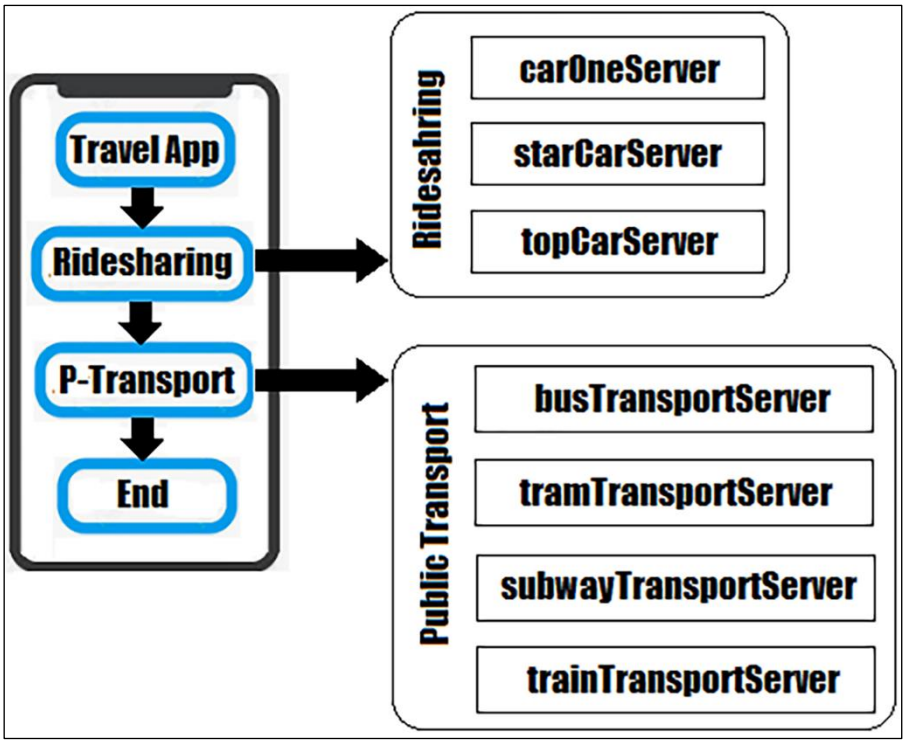
Chapter 11: Concurrency – Virtual Threads and Structured Concurrency: Diving Deeper

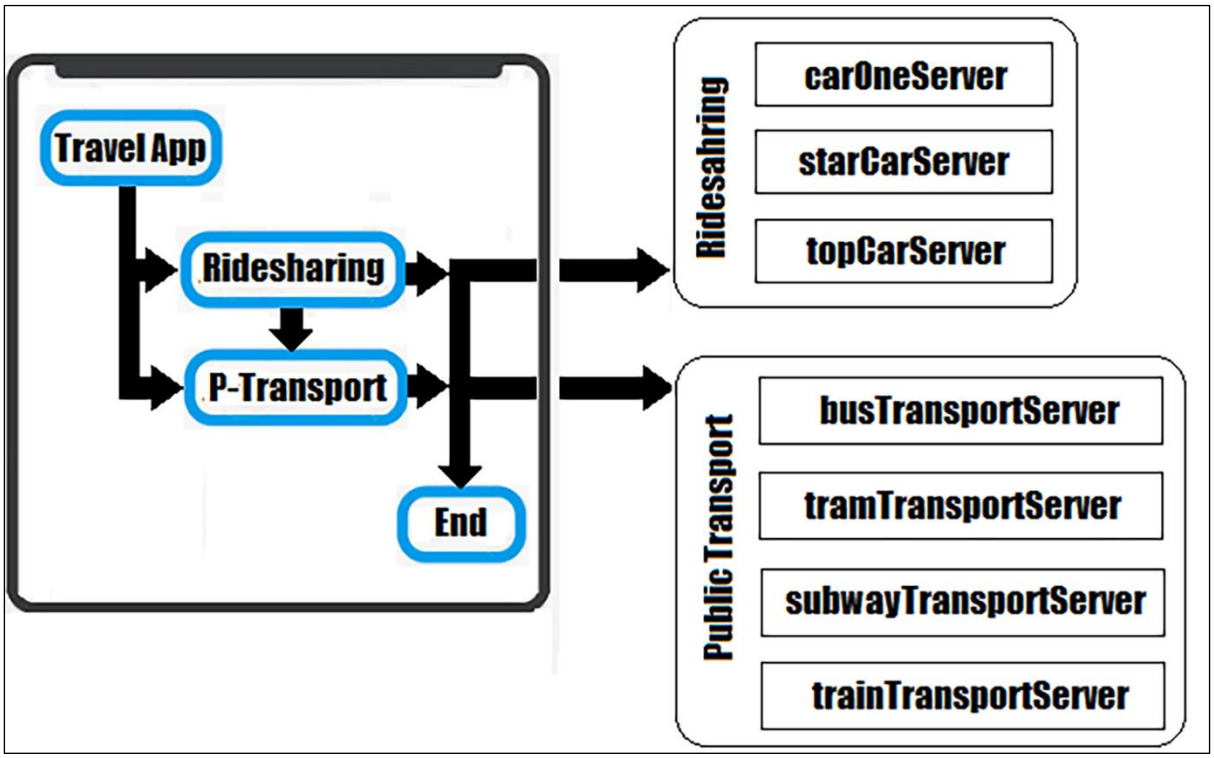





Thread
Thread
OS
Virtual Thread #41
Platform Thread worker-1
Virtual Thread #22
Platform Thread worker-4
ForkJoinPool
logger.info(...);
Thread.sleep(1000);
Continuation.run()
logger.info(...);
JVM
Heap


NEW
Thread.start()
STARTED
failed to start
TERMINATED
first run
job done
PARKING
parking attempt
RUNNING
Thread.yield()
YIELDING
yield failed
yield() succeed
yield() failed
unpark interrupted
resume execution
yield() succeed
PARKED
PINNED
RUNNABLE
Not mounted
Mounted
Transition
unpark interrupted


Travel App
Ridesharing
P-Transport
End
Ridesahring
carOneServer
starCarServer
topCarServer
Public Transport
busTransportServer
tramTransportServer
subwayTransportServer
trainTransportServer

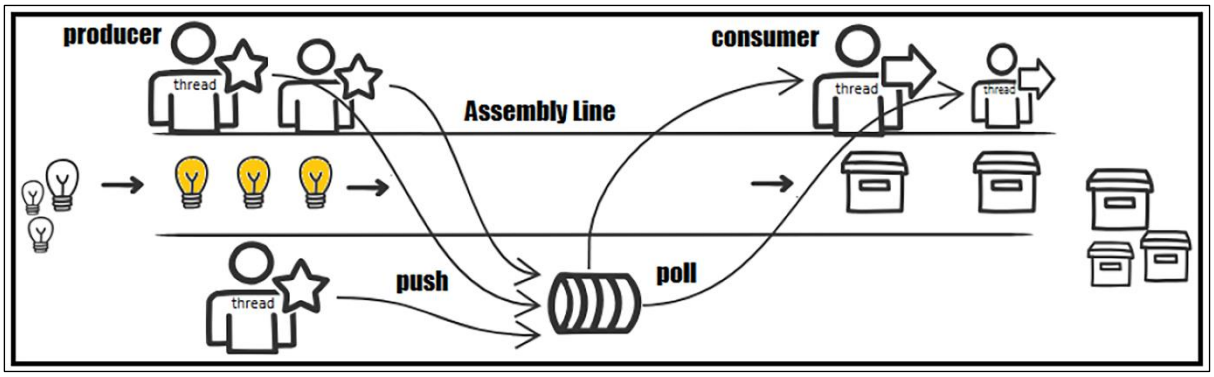

Travel App
Ridesharing
P-Transport
End
Ridesahring
carOneServer
starCarServer
topCarServer
Public Transport
busTransportServer
tramTransportServer
subwayTransportServer
trainTransportServer

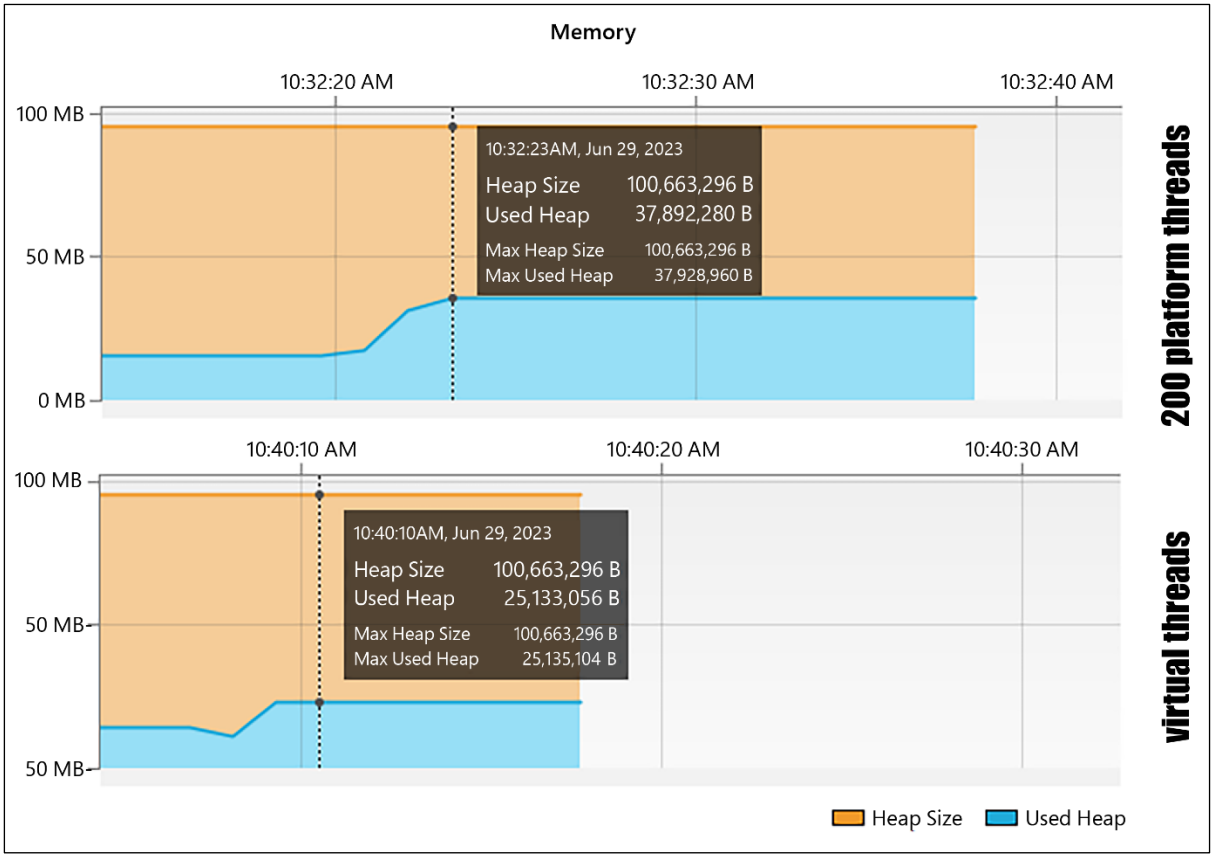
```java
public static void sayHelloTL() {

 logger.info(Thread.currentThread().toString());
 logger.info(() -> "Hello " + threadLocal.get());
}

public static void sayGoodByeTL() {

 logger.info(Thread.currentThread().toString());
 logger.info(() -> "Good bye " + threadLocal.get());
}

threadLocal.set("Mike");
sayHelloTL();
sayGoodByeTL();

------------------------
Hello Mike
Good bye Mike
```

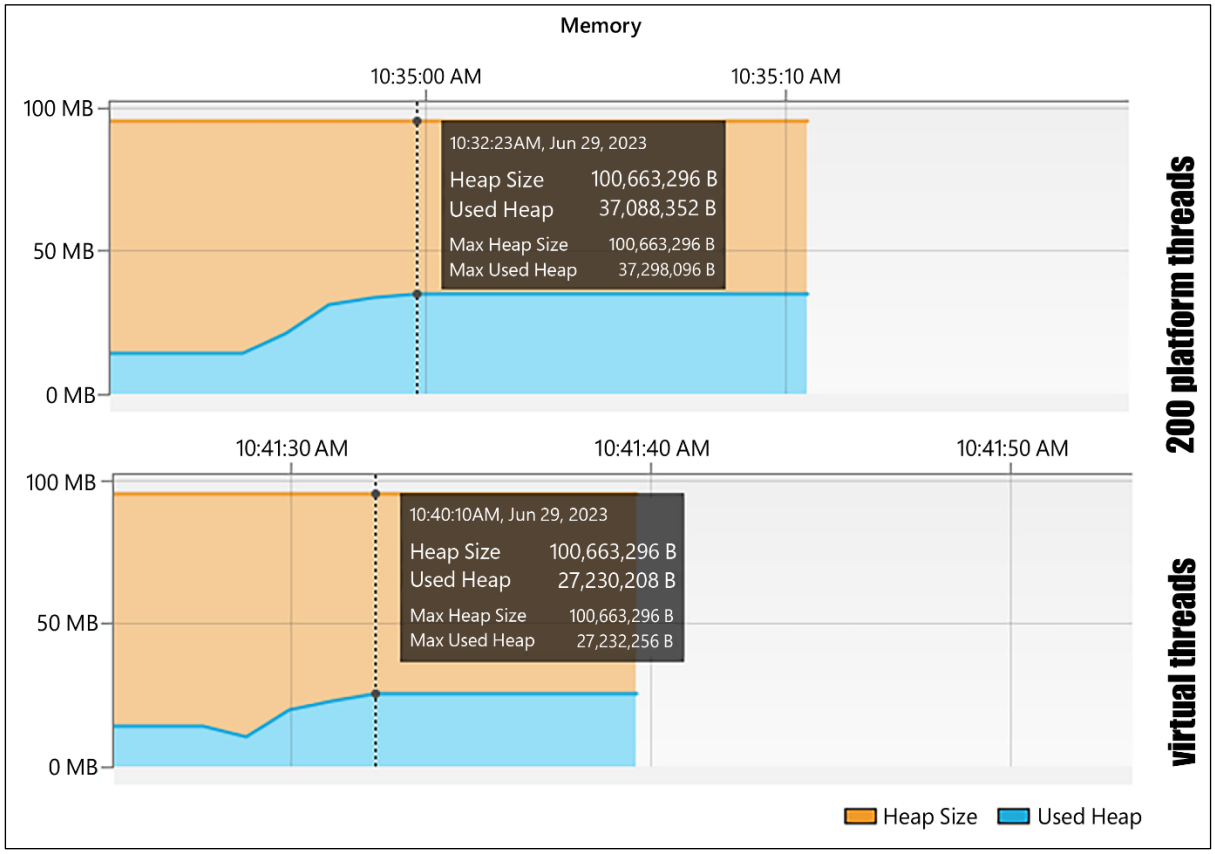
ThreadLocal

```java
public static void sayHelloSV() {

 logger.info(Thread.currentThread().toString());
 logger.info(() -> "Hello " + SCOPED_VALUE.orElse("you"));
}

public static void sayGoodByeSV() {

 logger.info(Thread.currentThread().toString());
 logger.info(() -> "Good bye " + SCOPED_VALUE.orElse("you"));
}

ScopedValue.where(SCOPED_VALUE, "Mike").run(() -> sayHelloSV());
sayGoodByeSV();


----------------------
Hello Mike
Good bye you
```

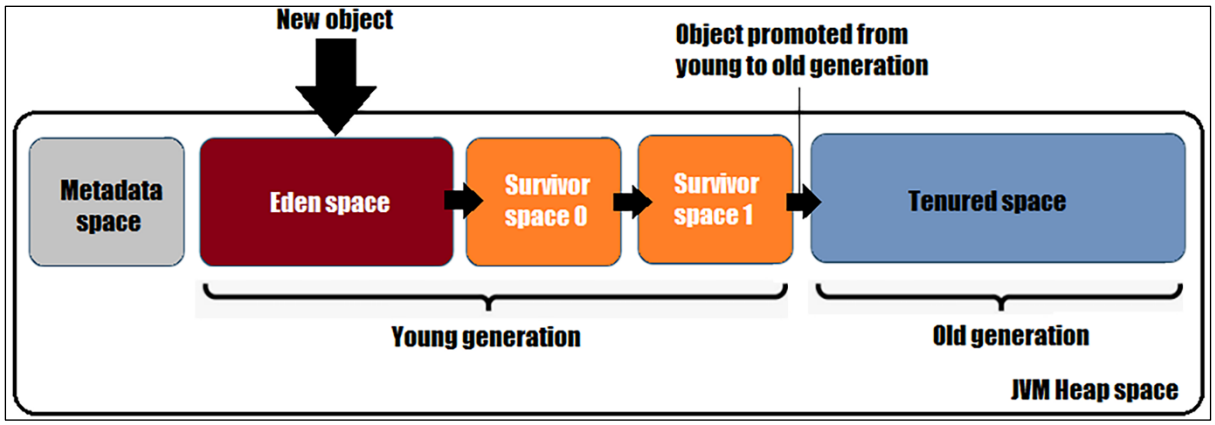
ScopedValue

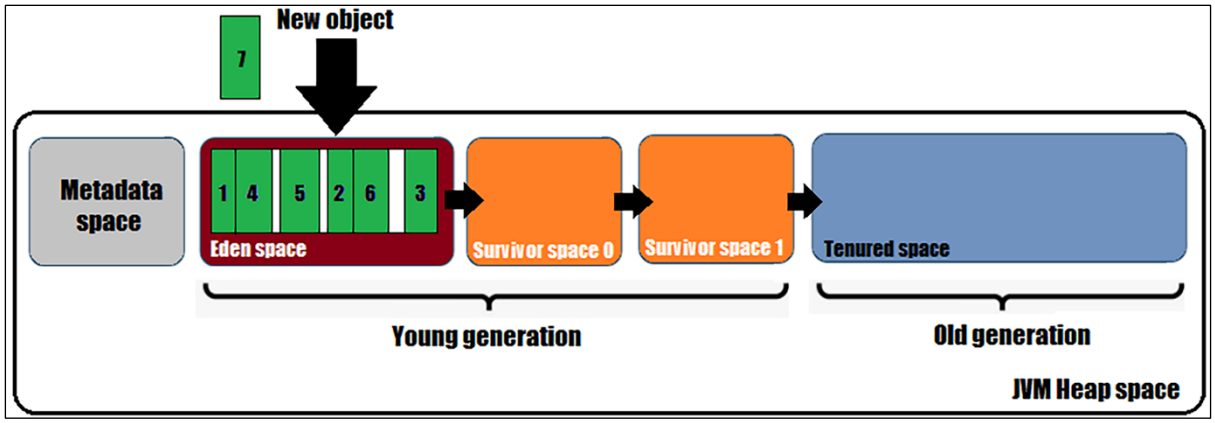


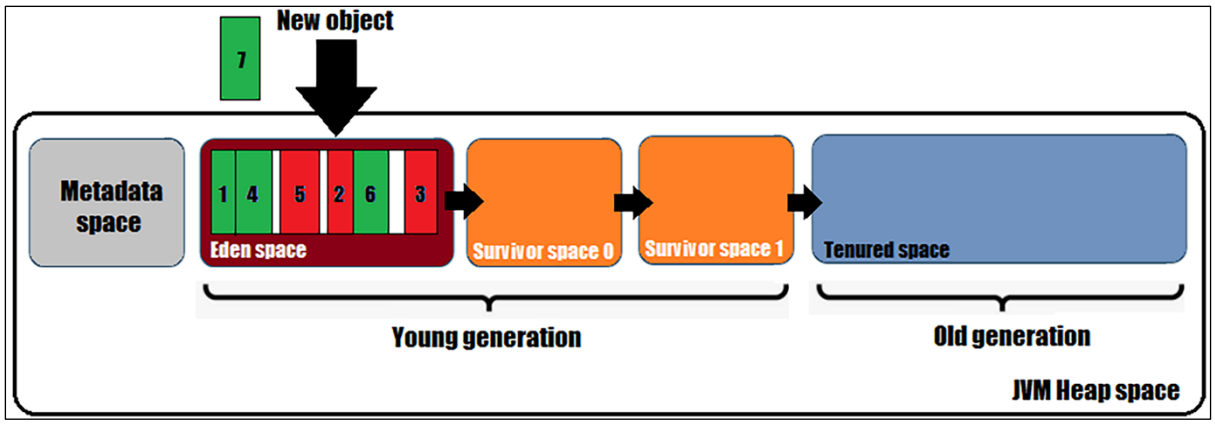


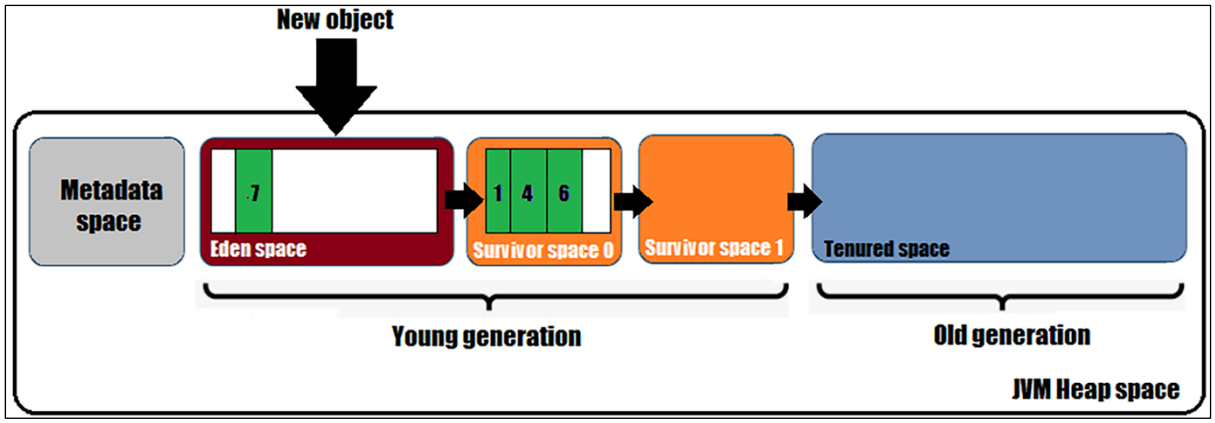

Memory
10:35:00 AM
10:35:10 AM
100 MB
50 MB
0 MB
10:32:23AM, Jun 29, 2023
Heap Size 100,663,296 B
Used Heap 37,088,352 B
Max Heap Size 100,663,296 B
Max Used Heap 37,298,096 B
200 platform threads
10:41:30 AM
10:41:40 AM
10:41:50 AM
100 MB
50 MB
0 MB
10:40:10AM, Jun 29, 2023
Heap Size 100,663,296 B
Used Heap 27,230,208 B
Max Heap Size 100,663,296 B
Max Used Heap 27,232,256 B
virtual threads
Heap Size
Used Heap

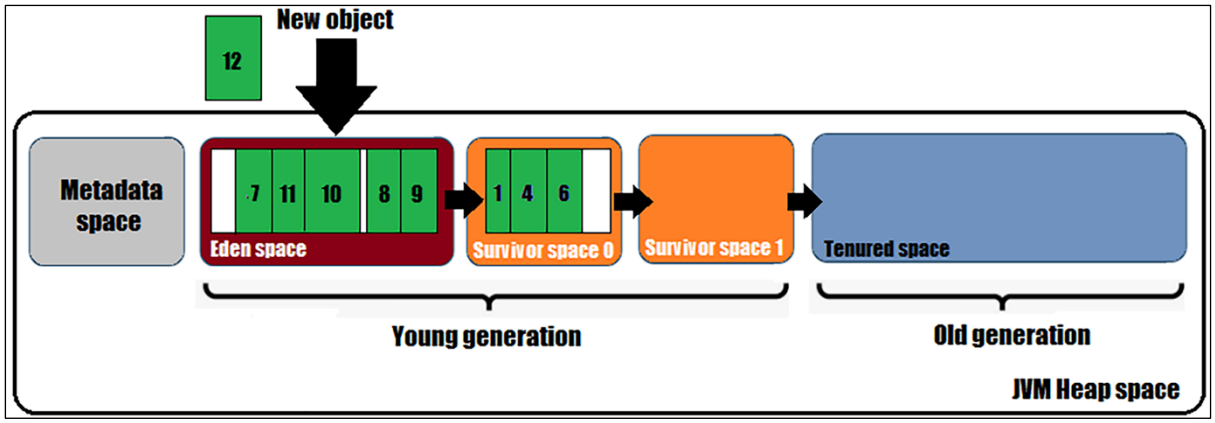
Chapter 12: Garbage Collectors and Dynamic CDS Archives

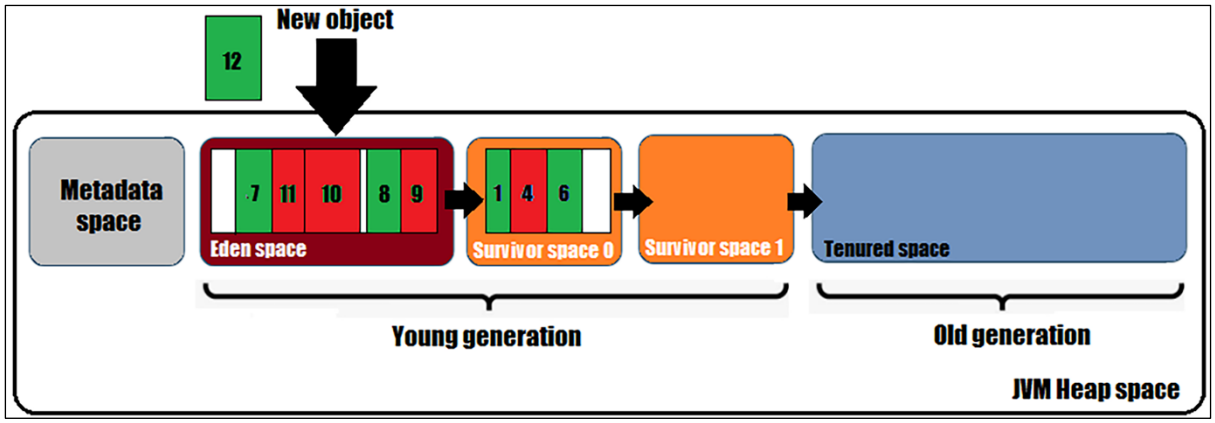


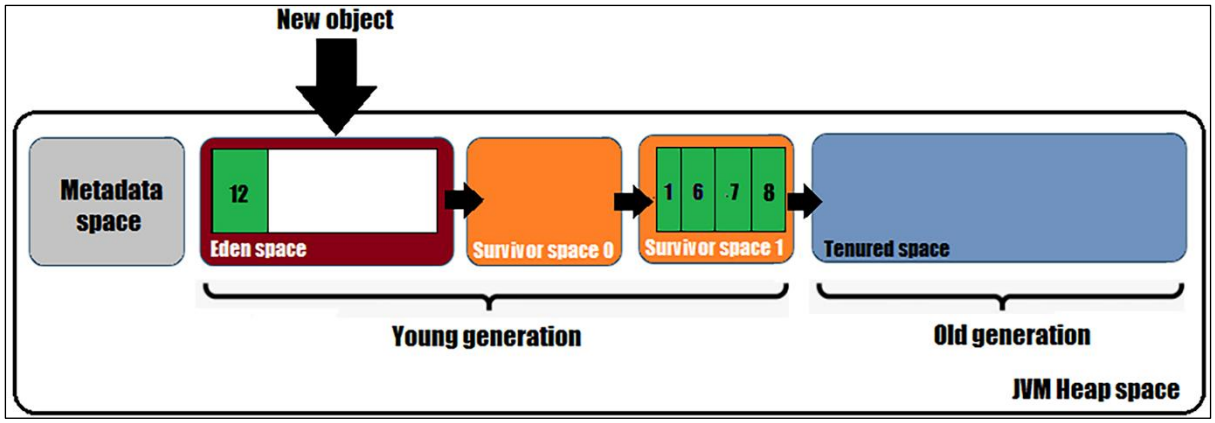


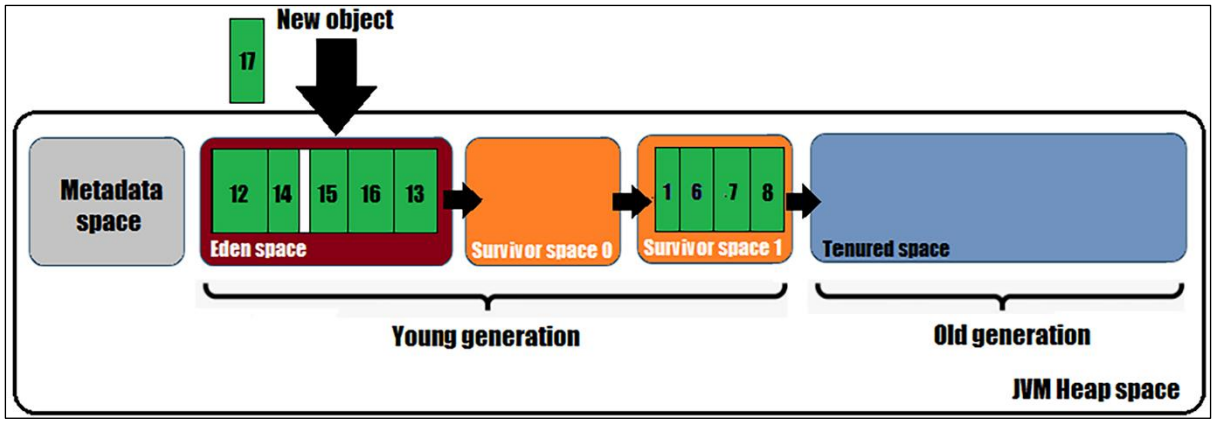


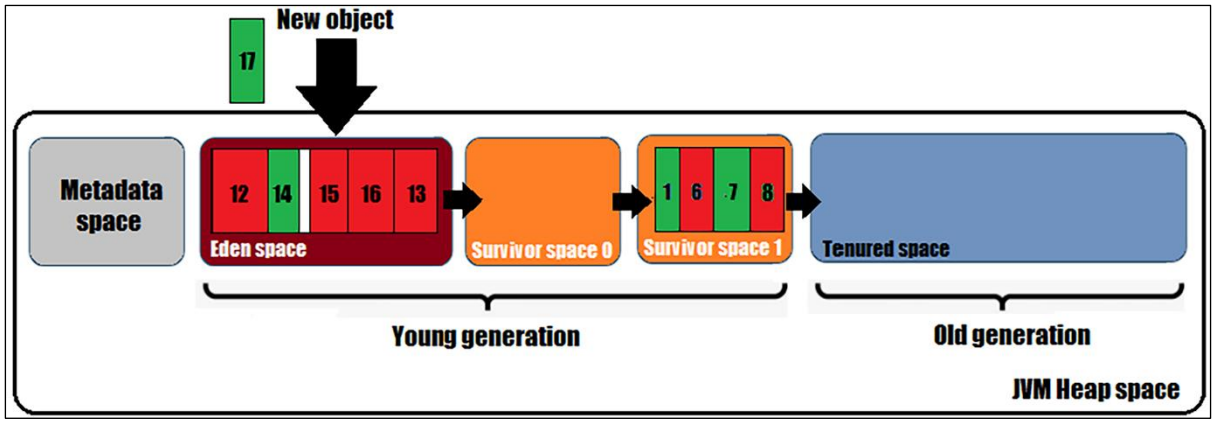

New object
Metadata space
7
Eden space
1
4
6
Survivor space 0
Survivor space 1
Tenured space
Young generation
Old generation
JVM Heap space

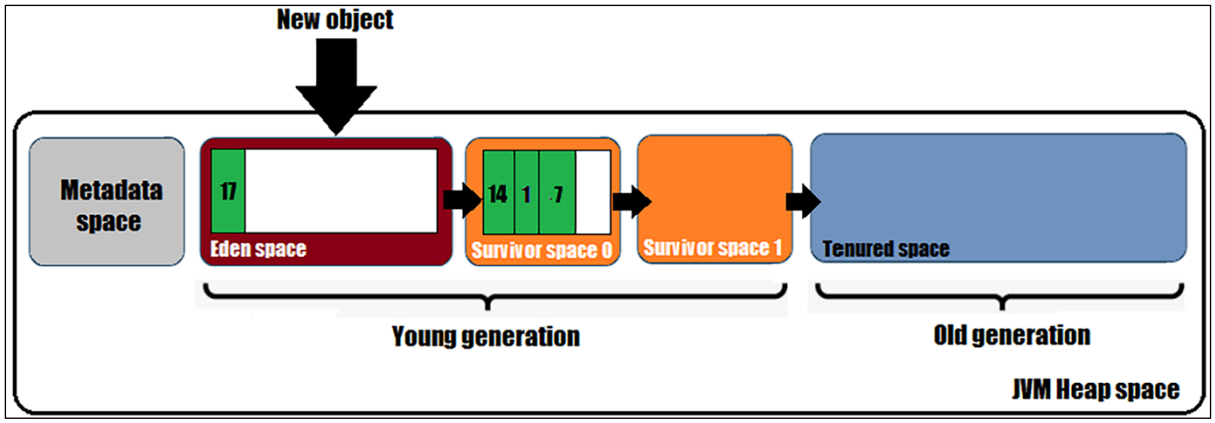

12
New object
Metadata space
7
11
10
8
9
Eden space
1
4
6
Survivor space 0
Survivor space 1
Tenured space
Young generation
Old generation
JVM Heap space

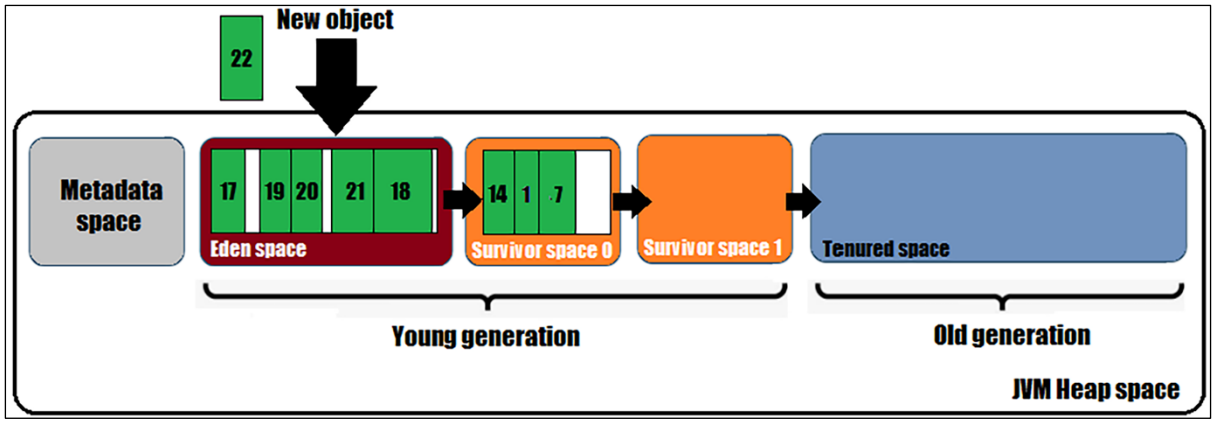

12
New object
Metadata space
7
11
10
8
9
Eden space
1
4
6
Survivor space 0
Survivor space 1
Tenured space
Young generation
Old generation
JVM Heap space

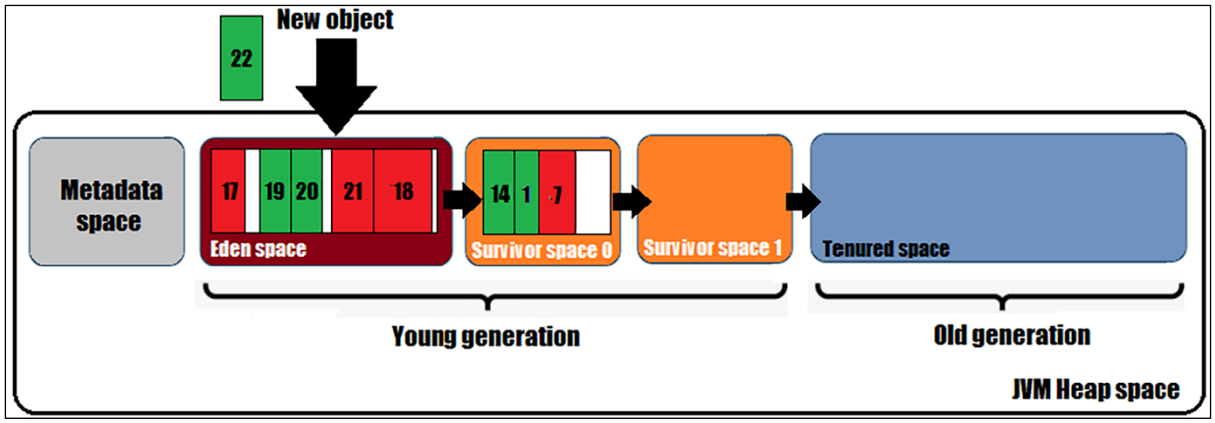

New object
Metadata space
12
Eden space
Survivor space 0
1
6
7
8
Survivor space 1
Tenured space
Young generation
Old generation
JVM Heap space

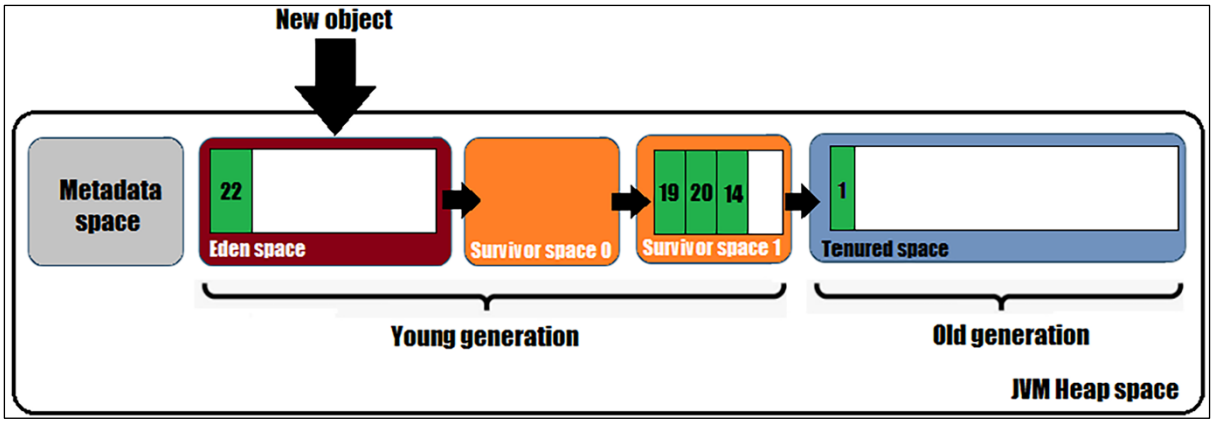

17
New object
Metadata space
12
14
15
16
13
Eden space
Survivor space 0
1
6
7
8
Survivor space 1
Tenured space
Young generation
Old generation
JVM Heap space

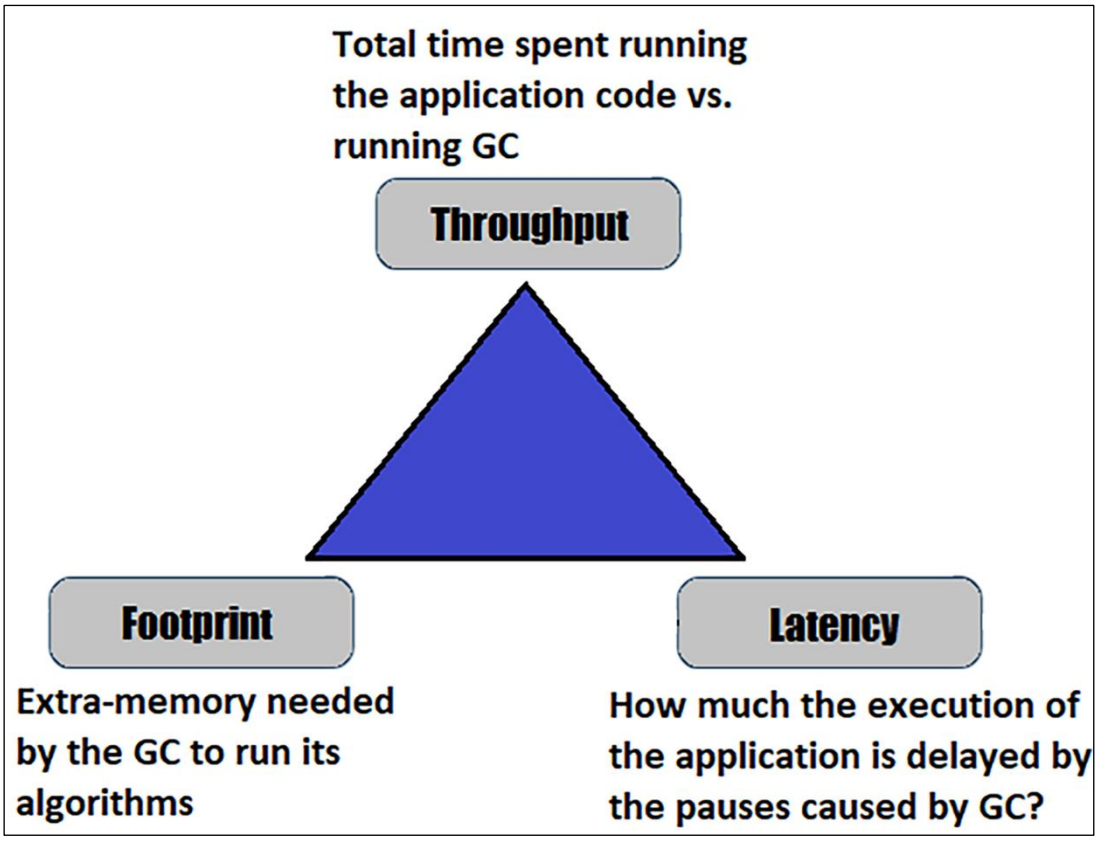

17
New object
Metadata space
12
14
15
16
13
Eden space
Survivor space 0
1
6
7
8
Survivor space 1
Tenured space
Young generation
Old generation
JVM Heap space

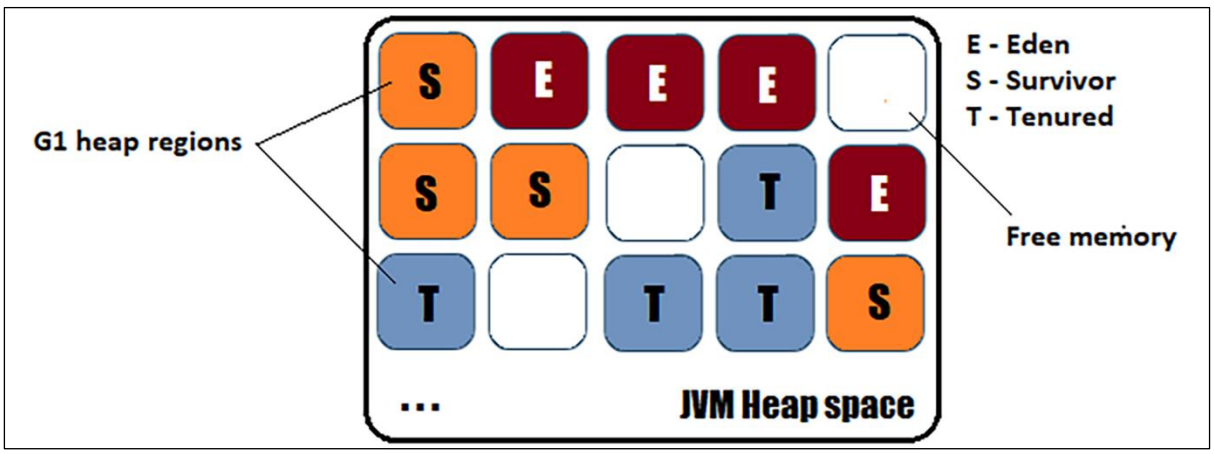

New object
Metadata space
17
Eden space
14
1
7
Survivor space 0
Survivor space 1
Tenured space
Young generation
Old generation
JVM Heap space

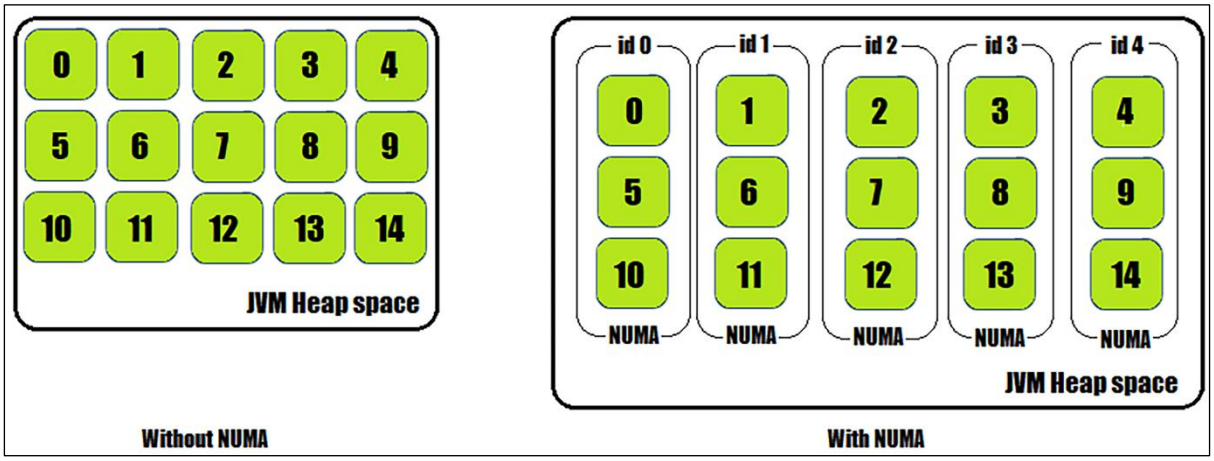

22
New object
Metadata space
17
19
20
21
18
Eden space
14
1
7
Survivor space 0
Survivor space 1
Tenured space
Young generation
Old generation
JVM Heap space

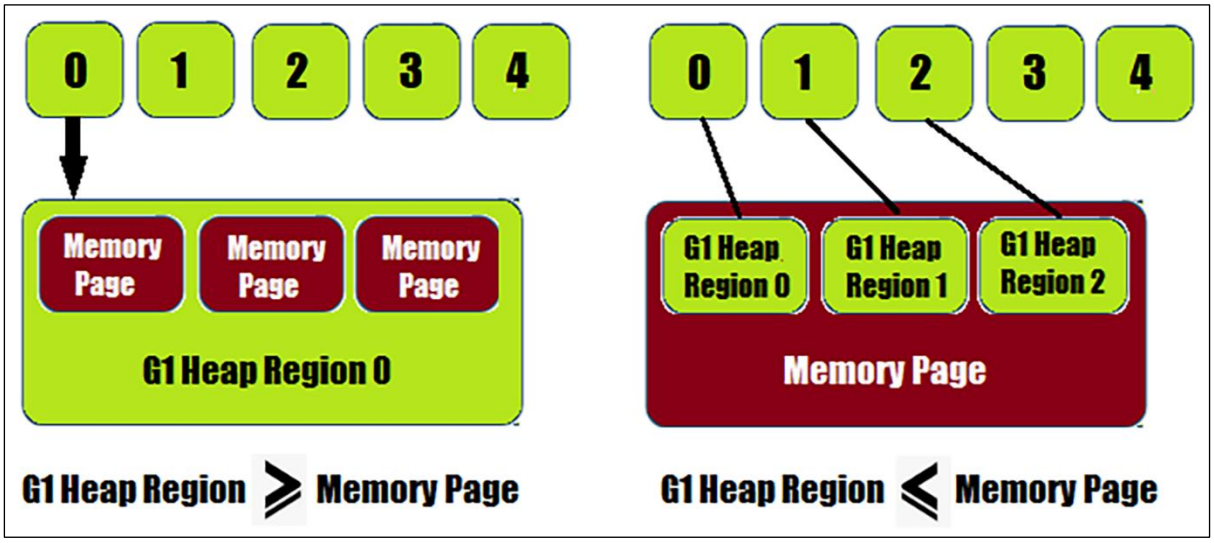

22
New object
Metadata space
17
19
20
21
18
Eden space
14
1
7
Survivor space 0
Survivor space 1
Tenured space
Young generation
Old generation
JVM Heap space

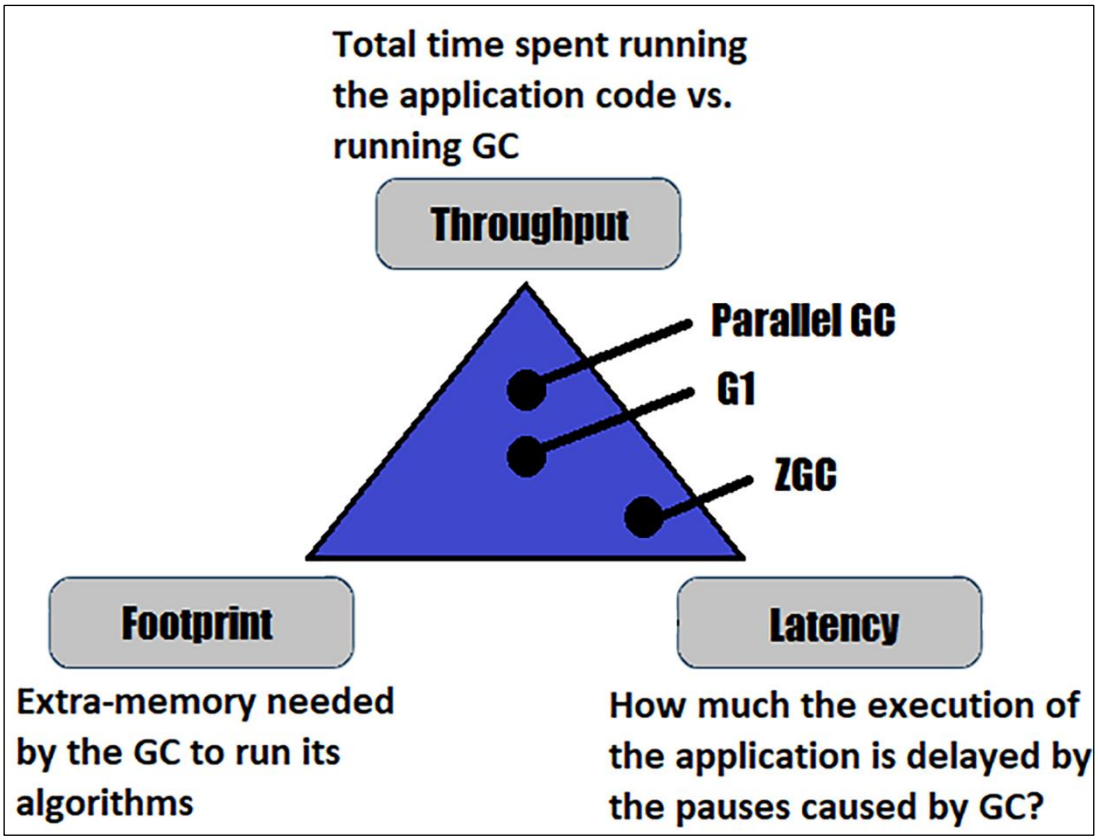

New object
Metadata space
22
Eden space
Survivor space 0
19 20 14
Survivor space 1
1
Tenured space
Young generation
Old generation
JVM Heap space

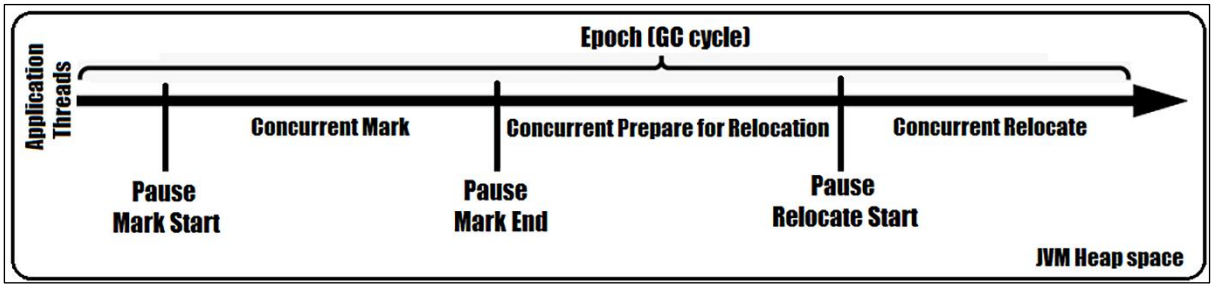

Total time spent running the application code vs. running GC
Throughput
Footprint
Latency
Extra-memory needed by the GC to run its algorithms
How much the execution of the application is delayed by the pauses caused by GC?

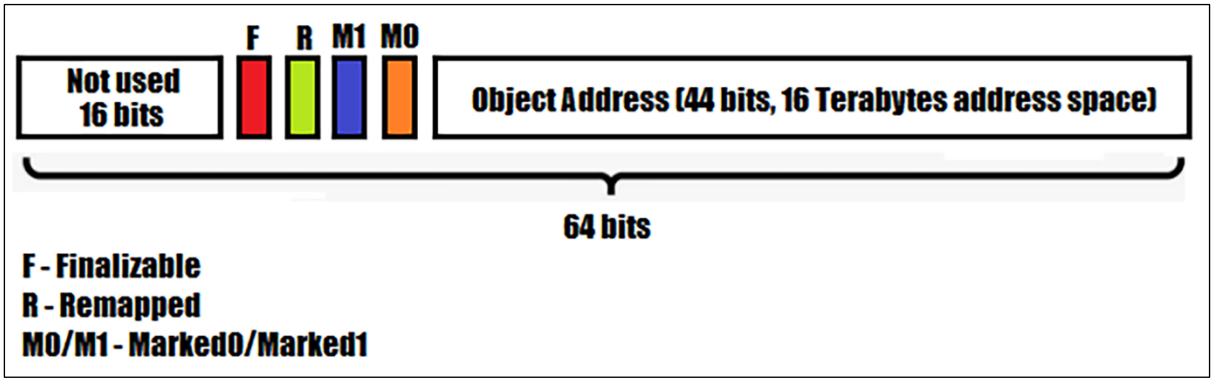

G1 heap regions
S E E E
S S T E
T T T S
...
JVM Heap space
E - Eden
S - Survivor
T - Tenured
Free memory

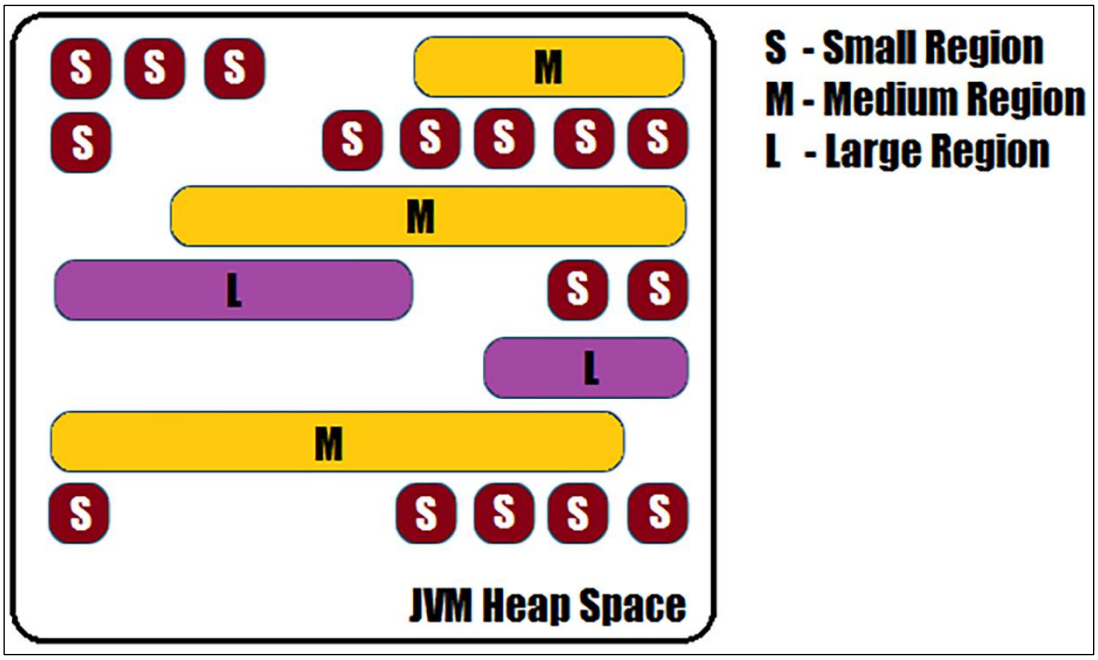

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
JVM Heap space
Without NUMA
id 0
id 1
id 2
id 3
id 4
NUMA
JVM Heap space
With NUMA

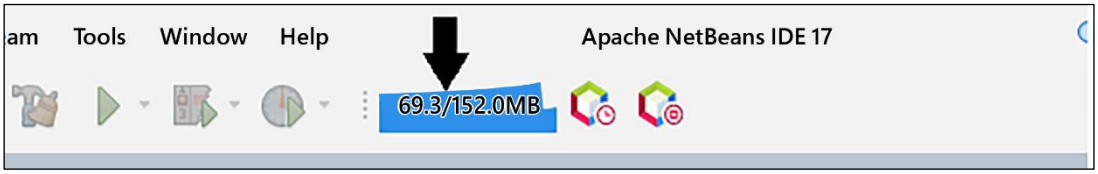

0
1
2
3
4
Memory Page
Memory Page
Memory Page
G1 Heap Region 0
G1 Heap Region ≥ Memory Page
G1 Heap Region 0
G1 Heap Region 1
G1 Heap Region 2
Memory Page
G1 Heap Region ≤ Memory Page

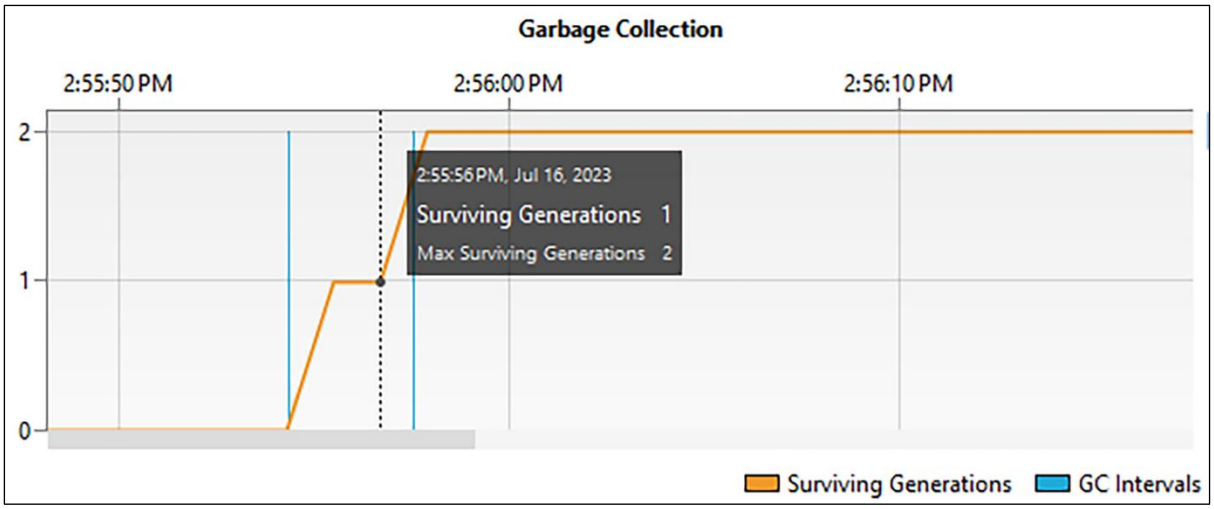

Total time spent running the application code vs. running GC
Throughput
Parallel GC
G1
ZGC
Footprint
Latency
Extra-memory needed by the GC to run its algorithms
How much the execution of the application is delayed by the pauses caused by GC?

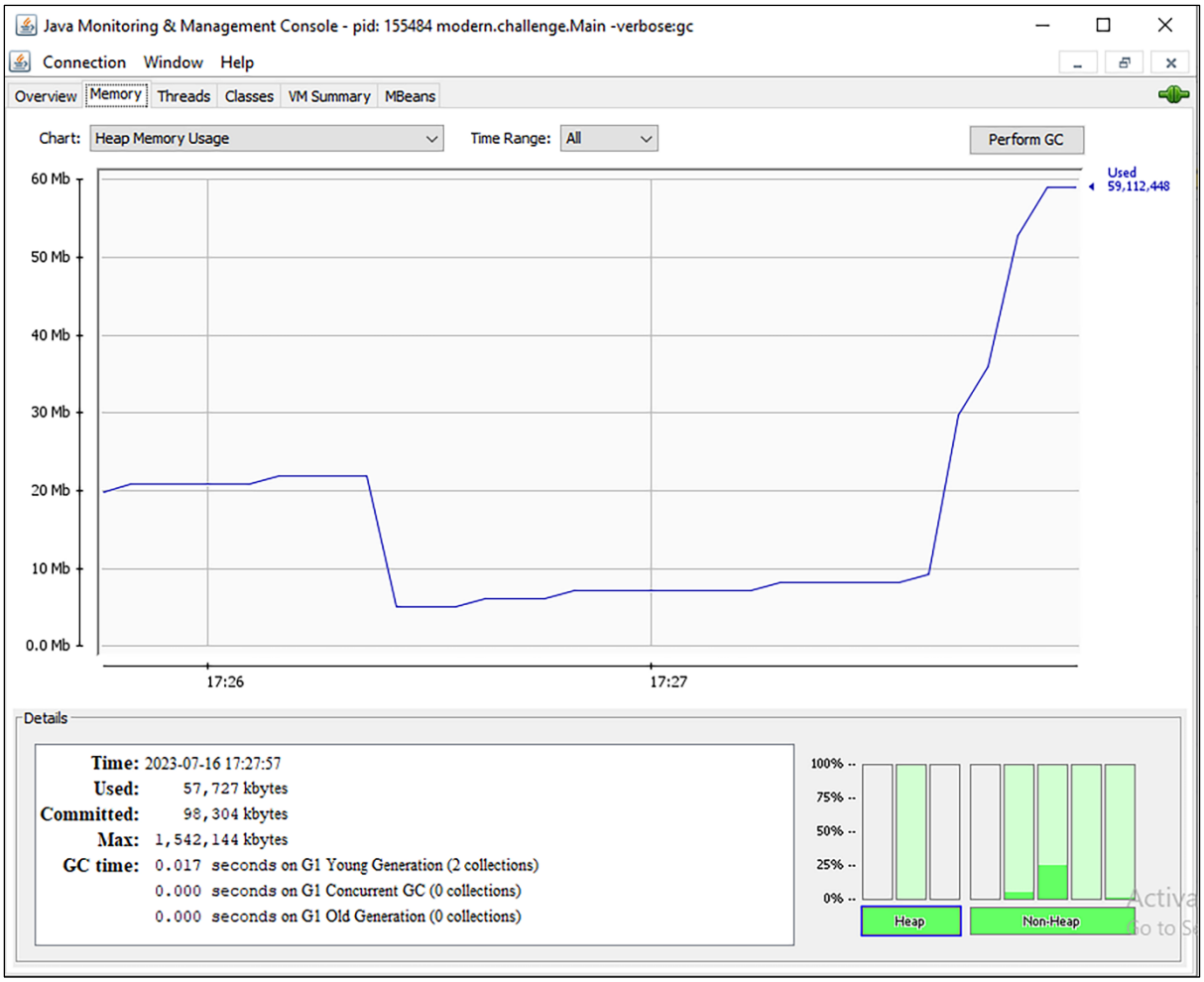

Epoch (GC cycle)
Application Threads
Concurrent Mark
Concurrent Prepare for Relocation
Concurrent Relocate
Pause Mark Start
Pause Mark End
Pause Relocate Start
JVM Heap space


F
R
M1
M0
Not used 16 bits
Object Address (44 bits, 16 Terabytes address space)
64 bits
F - Finalizable
R - Remapped
M0/M1 - Marked0/Marked1

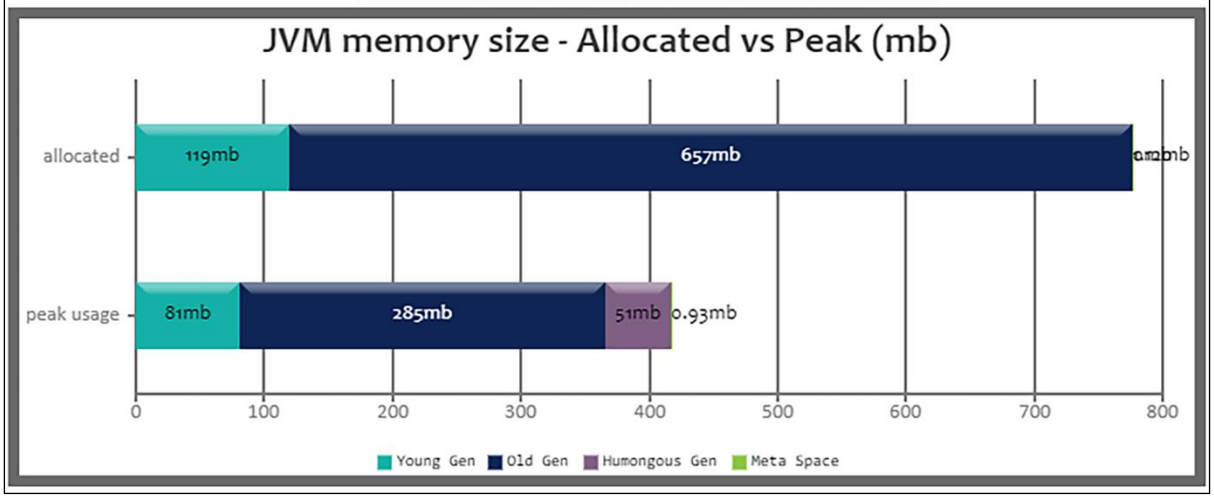

S - Small Region
M - Medium Region
L - Large Region
JVM Heap Space

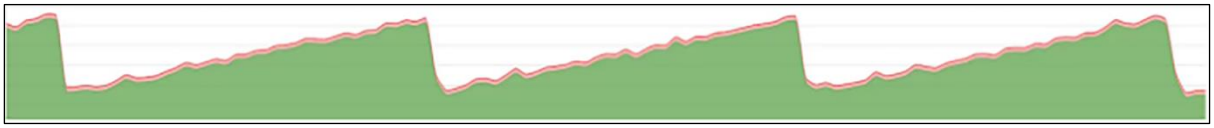

am
Tools
Window
Help
Apache NetBeans IDE 17
69.3/152.0MB


Garbage Collection
2:55:50 PM
2:56:00 PM
2:56:10 PM
2
1
0
2:55:56 PM, Jul 16, 2023
Surviving Generations 1
Max Surviving Generations 2
Surviving Generations
GC Intervals

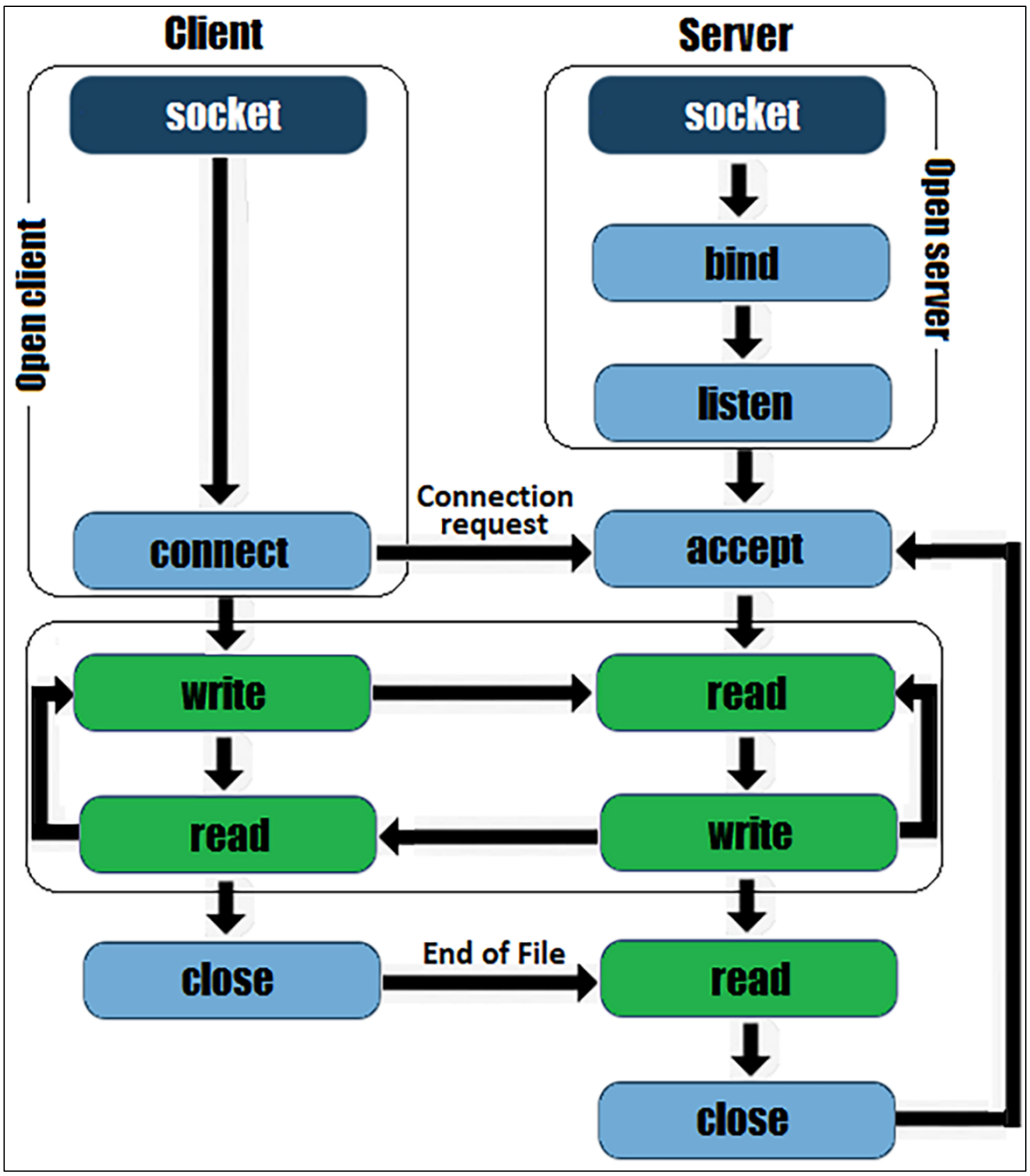

Java Monitoring & Management Console - pid: 155484 modern.challenge.Main -verbose:gc
Connection Window Help
Overview Memory Threads Classes VM Summary MBeans
Chart: Heap Memory Usage
Time Range: All
Perform GC
60 Mb
50 Mb
40 Mb
30 Mb
20 Mb
10 Mb
0.0 Mb
17:26
17:27
Used
59,112,448
Details
Time: 2023-07-16 17:27:57
Used: 57,727 kbytes
Committed: 98,304 kbytes
Max: 1,542,144 kbytes
GC time: 0.017 seconds on G1 Young Generation (2 collections)
0.000 seconds on G1 Concurrent GC (0 collections)
0.000 seconds on G1 Old Generation (0 collections)
100%
75%
50%
25%
0%
Heap
Non-Heap

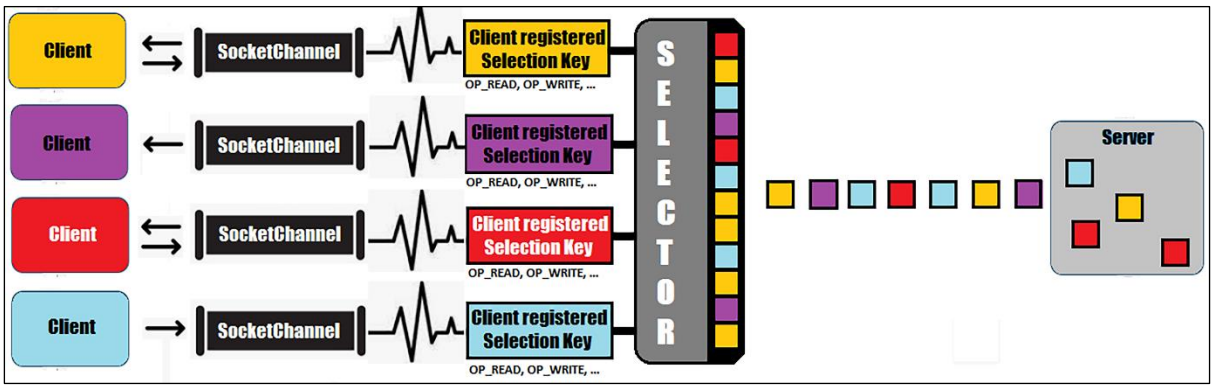
Generation	Allocated	Peak
Young Generation	119 mb	81 mb
Old Generation	657 mb	285 mb
Humongous	n/a	51 mb
Meta Space	1.12 mb	950 kb
Young + Old + Meta space	777.12 mb	369.93 mb

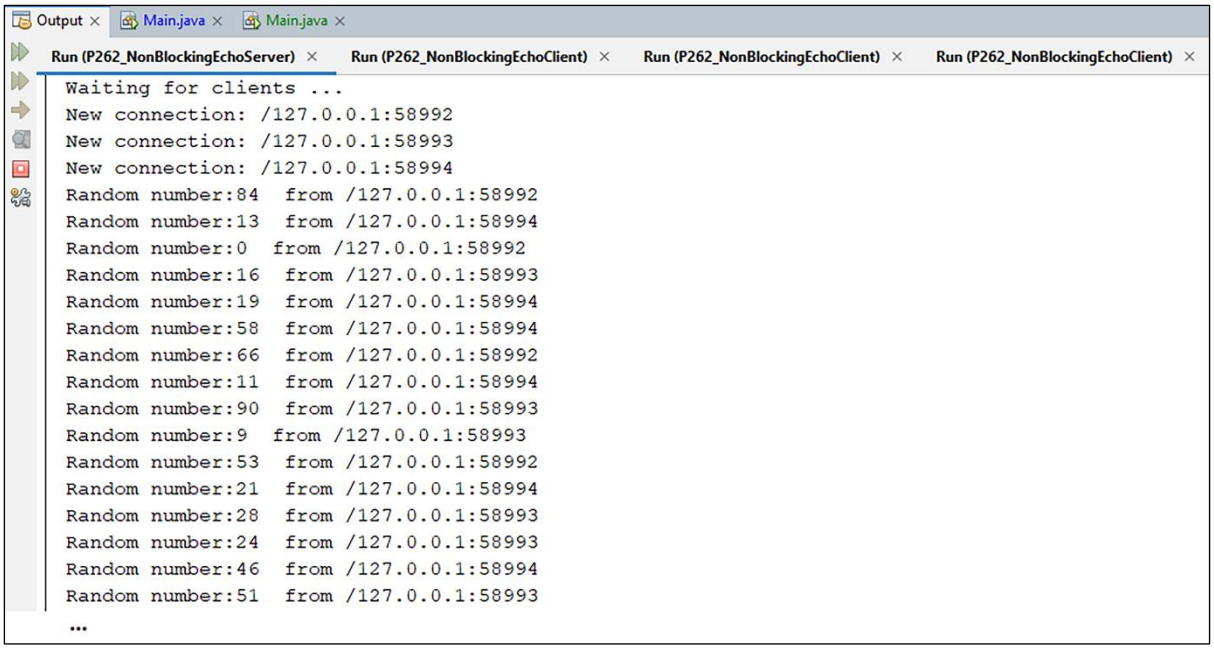




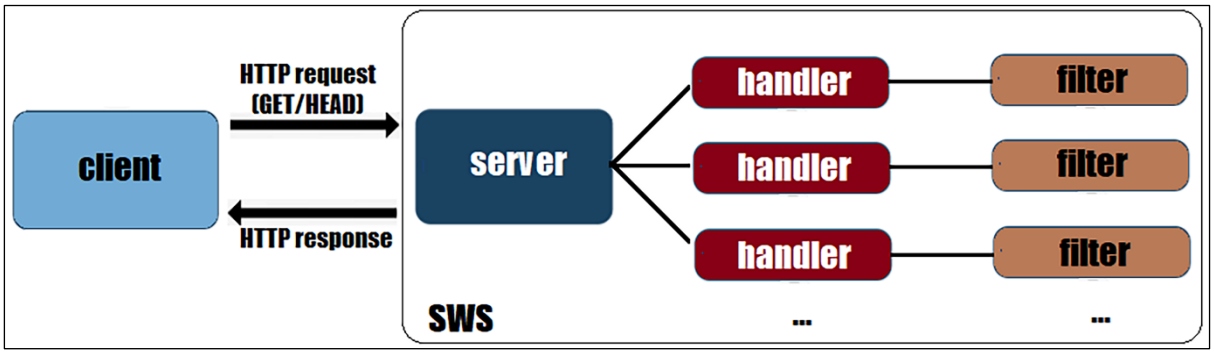
Chapter 13: Socket API and Simple Web Server

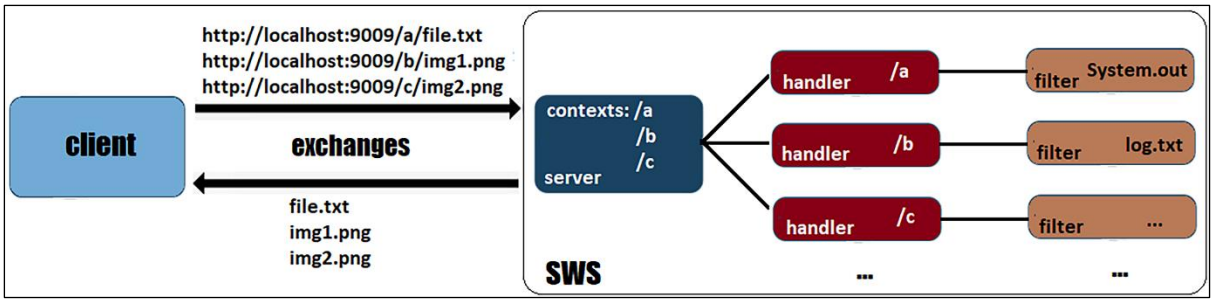







```
Display Name: Hyper-V Virtual Ethernet Adapter
Hyper-V Virtual Ethernet Adapter is up and running ? true
Hyper-V Virtual Ethernet Adapter is multicast capable ? true
Hyper-V Virtual Ethernet Adapter name: ethernet_32775
Hyper-V Virtual Ethernet Adapter is virtual ? false
IP addresses:
IP: /fe80:0:0:0:4599:d8a1:418c:2427%ethernet_32775
IP: /172.18.64.1
```






```
jwebserver [-b bind address] [-p port] [-d directory]
           [-o none|info|verbose] [-h to show options]
           [-version to show version information]
```

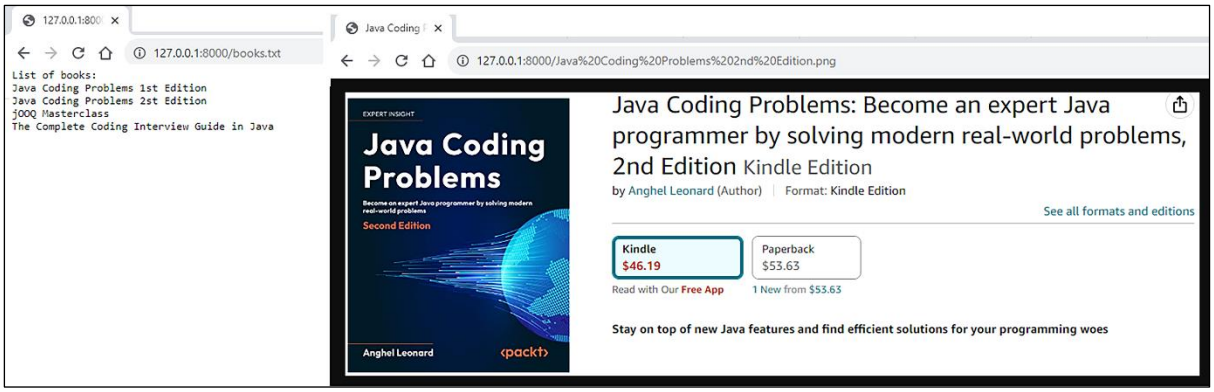
```
Command Prompt - jwebserver

C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition\Chapter13\P270_SWSCommandLineTool\docs>jwebserver
Binding to loopback by default. For all interfaces use "-b 0.0.0.0" or "-b ::".
Serving C:\SBPBP\GitHub\Java-Coding-Problems-Second-Edition
         \Chapter13\P270_SWSCommandLineTool\docs and subdirectories on 127.0.0.1 port 8000
URL http://127.0.0.1:8000/
```

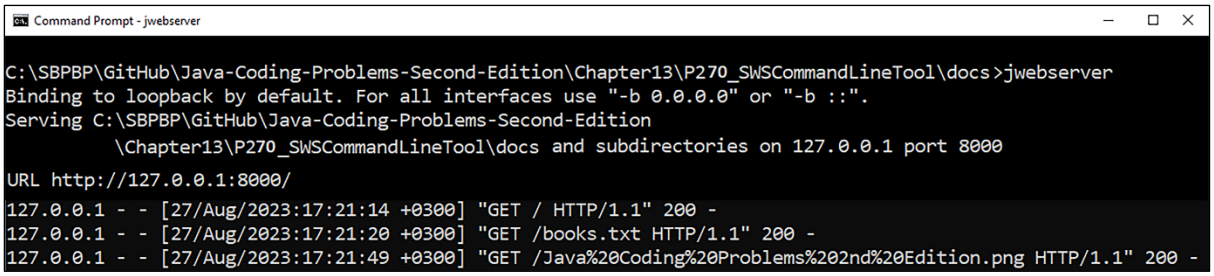
127.0.0.1:8000

127.0.0.1:8000

Directory listing for /

- books.txt
- Java Coding Problems 1st Edition.png
- Java Coding Problems 2nd Edition.png
- jOOQ Masterclass.png
- The Complete Coding Interview Guide in Java.png






```
> Accept-language: en-US,en;q=0.9,ro;q=0.8
> Upgrade-insecure-requests: 1
> Author: Anghel Leonard
> If-modified-since: Tue, 29 Aug 2023 11:02:50 GMT
> Sec-ch-ua-mobile: ?0
> Cache-control: max-age=0
```