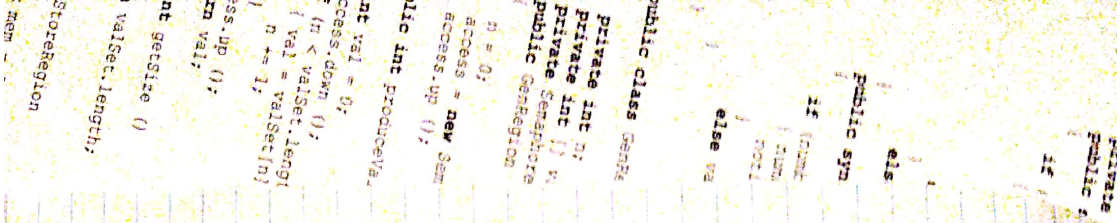
3) O valor processado por (val / 10000) e por ((val - 10000 * (Val / 10000)) / 100) por val % 100

$$\frac{\left(val - \left(10000 \times \frac{Val}{10000}\right)\right)}{100} = \frac{Val - val}{100} = \frac{0}{100}$$

$$= 0$$

Logo vão ser impressos 6 valores

O valor processado por 12 e por 0 vai dar 8
O valor processado por 11 e por 0 vai dar 22
O valor processado por 5 e por 0 vai dar 22
O valor processado por 2 e por 0 vai dar 23
O valor processado por 4 e por 0 vai dar 20
O valor processado por 5 e por 0 vai dar 19



mem = 10000 × (tmp[0][ind[0]] / 100)

+ tmp[0][ind[0]] % 100

+ 100 × val(/100) + Val % 100

= 100$$ × tm[0][ind[0]]) + tmp[0][ind[0]] %

+ val + Val % 100

~~708~~ + 8 + 303 + 3

= ~~422~~ 70

~~7~~ 2 708 + 8 + 407 + 7 = 30

17222

~~604~~ + 100 × 509 + 4 + 609 + 9

= 9

306 → 217

410

512

217 × 100 + 11 + 306 + 6 = 23 21723

410 × 100 + 10 + 105 + 5 = 41124

512 × 100 + 12 + 709 + 7

```
    public StoreRegion (int mult, int size)
    {
       this.mult = mult;
       tmp = new int[mult][size];
       ind = new int[mult];
       access.up ();
       stat = new Semaphore[mult+2];
       for (int i = 0; i < mult + 2; i++)
       { if (i < mult) ind[i] = 0;
         stat[i] = new Semaphore ();
       }
       stat[mult+1].up ();
    }
    public void putVal (int val)
    {
       int m = (val % 100) % mult;
       stat[mult+1].down ();
       access.down ();
       if (ind[(m + 1) % mult] != 0)
          { ind[(m + 1) % mult] -= 1;
            mem = 10000 * (tmp[(m + 1) % mult][ind[(m + 1) % mult]] / 100) +
                  tmp[(m + 1) % mult][ind[(m + 1) % mult]] % 100 + 100 * (val / 100) +
                  val % 100;
            for (int i = 0; i <= mult; i++)
              stat[i].up ();
          }
          else { tmp[m][ind[m]] = val;
                 ind[m] += 1;
                 stat[mult+1].up ();
               }
       access.up ();
       stat[m].down ();
    }
    public int getVal ()
    {
       int val;
       stat[mult].down ();
       access.down ();
       val = mem;
       mem = 0;
       stat[mult+1].up ();
       access.up ();
       return val;
    }
}

public class Resource
{
   private int n;
   private Semaphore access = new Semaphore ();
   public Resource (int n)
   {
      this.n = n;
      access.up ();
   }
   public boolean printVal (int val)
   {
      access.down ();
      if ((n > 0) && (val != 0))
         { System.out.println ("O valor processado por " + (val/10000) + " e por " +
                               ((val-10000*(val/10000))/100) + " foi " + (val%100) + ".");
           n -= 1;
         }
      access.up ();
      return (n == 0);
   }
}
```