ORC WG - CRA Mondays

Anyone who still puts AI-generated code into circulation today has conditional intent to infringe the law
– how to limit or at least defer the risk

bitsea.

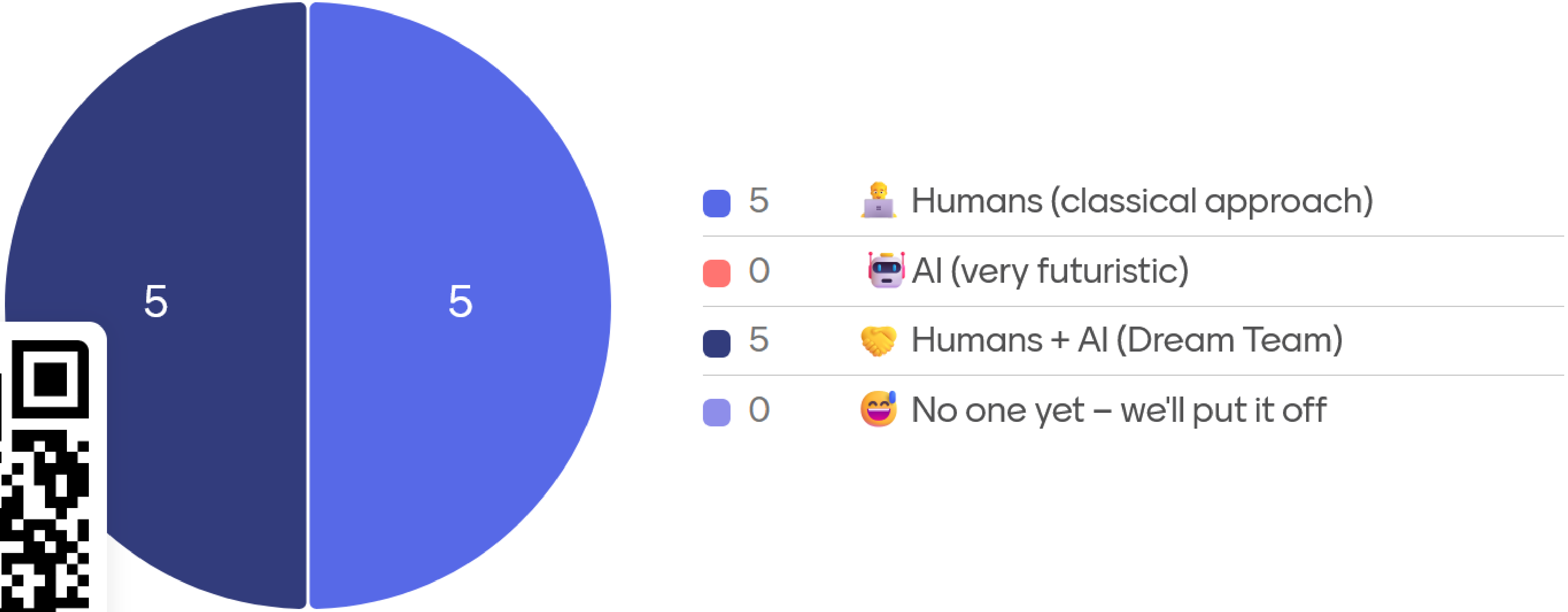
JUN

Dr. Andreas Kotulla
CEO Bitsea GmbH

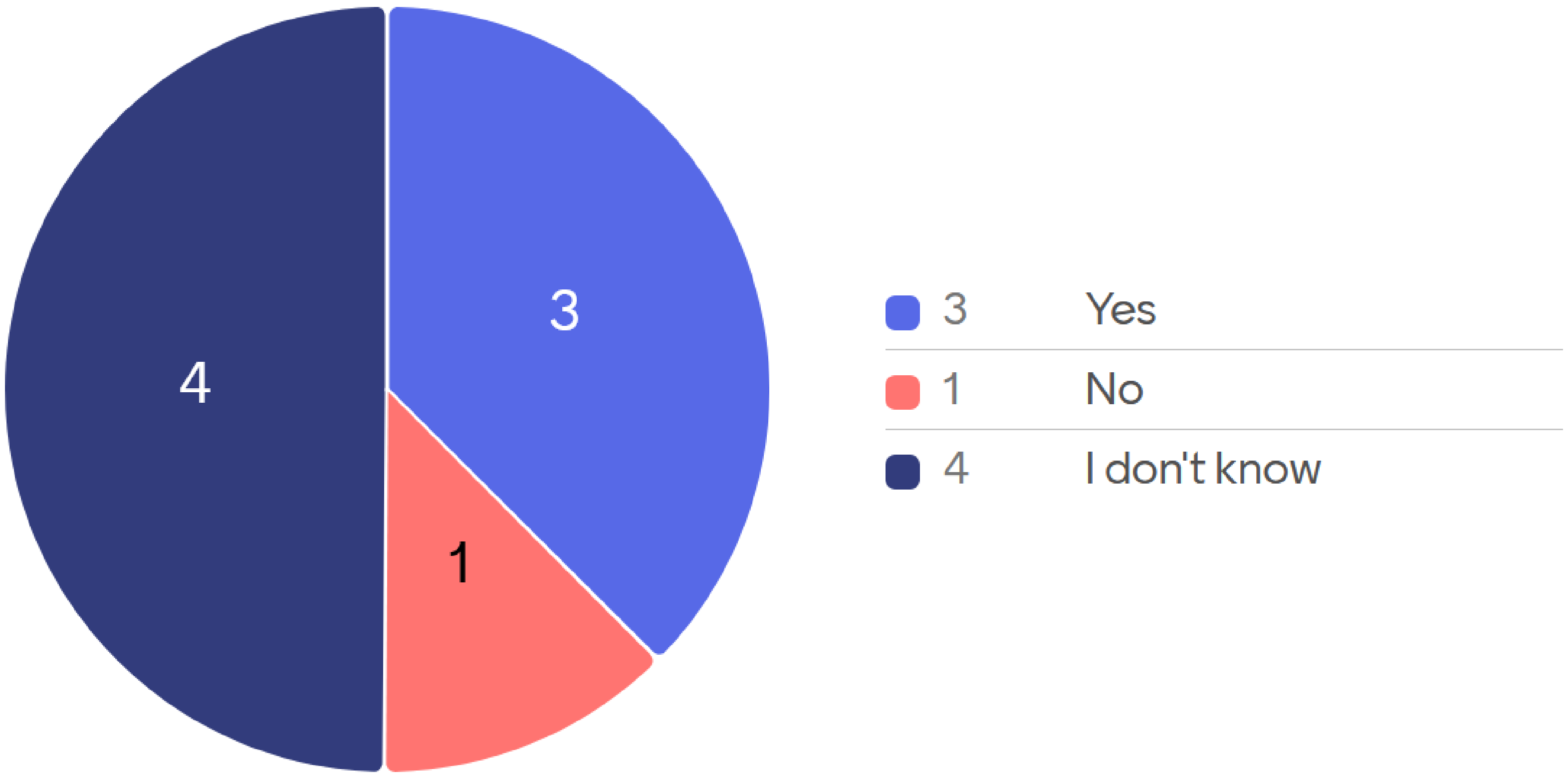
Florian Hackel
JUN Legal GmbH

February 2nd, 2026

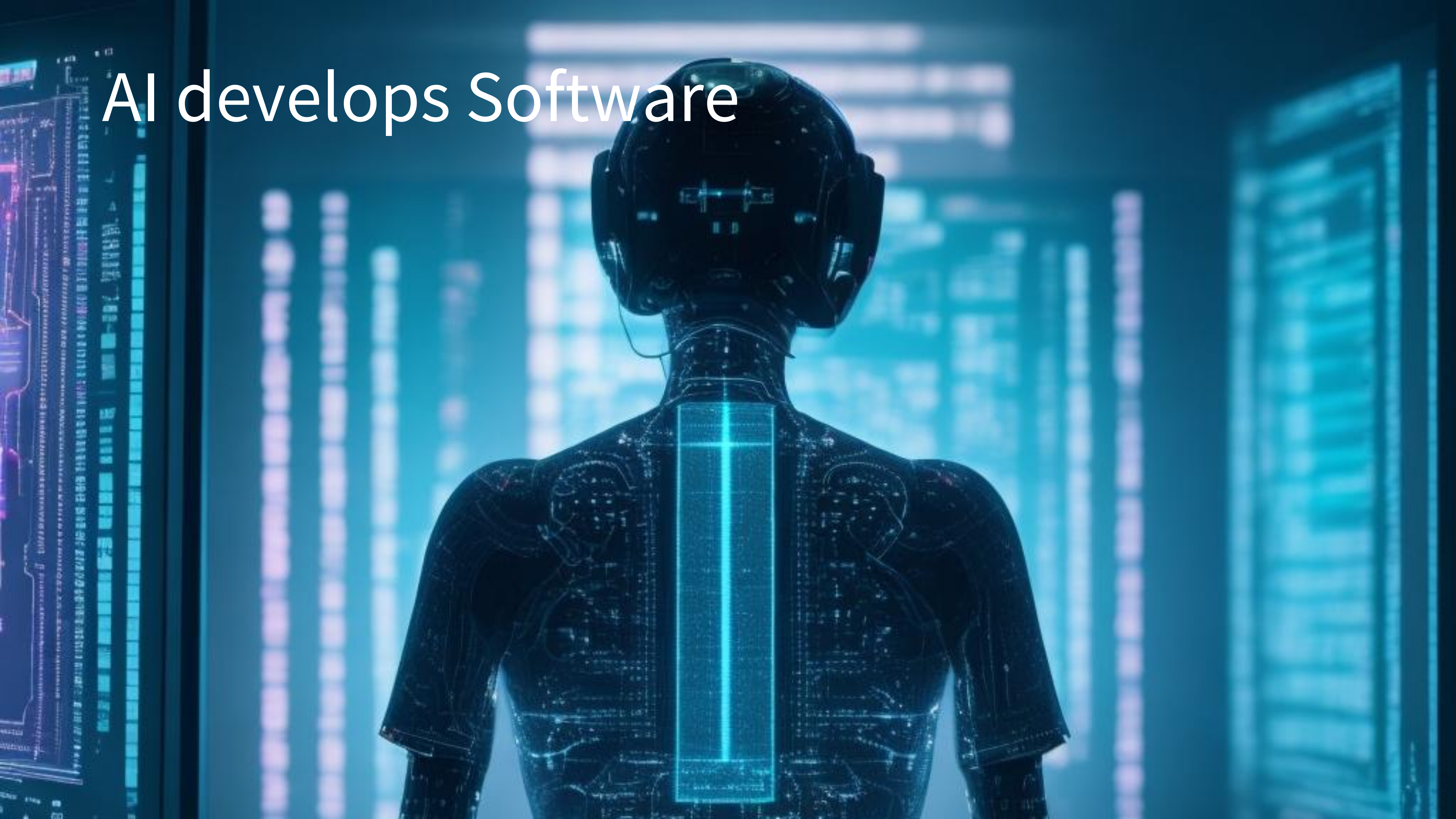
Who writes the code in your company?



Is the code being checked to ensure that it is legally sound?

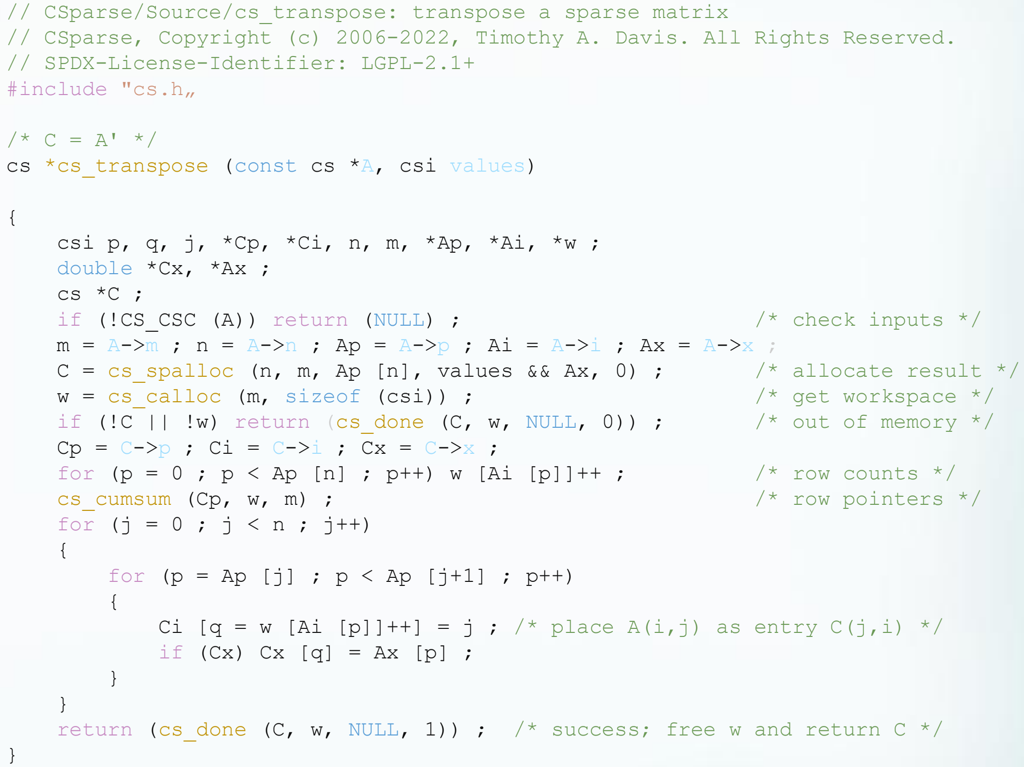


AI develops Software



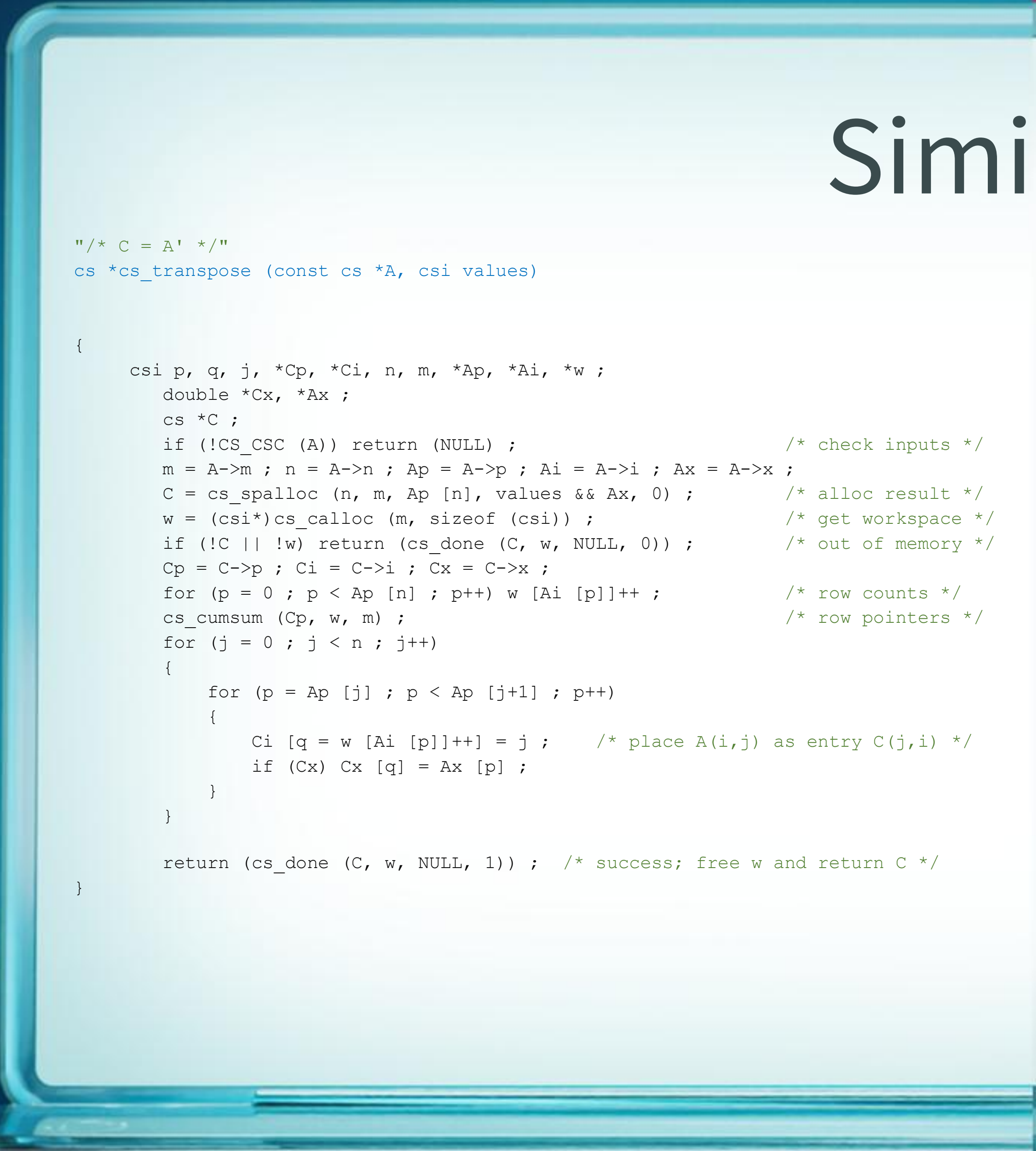
Similar?

```
"/* C = A' */"
cs *cs_transpose (const cs *A, csi values)


{
     csi p, q, j, *Cp, *Ci, n, m, *Ap, *Ai, *w ;
        double *Cx, *Ax ;
        cs *C ;
        if (!CS_CSC (A)) return (NULL) ;                    /* check inputs */
        m = A->m ; n = A->n ; Ap = A->p ; Ai = A->i ; Ax = A->x ;
        C = cs_spalloc (n, m, Ap [n], values && Ax, 0) ;     /* alloc result */
        w = (csi*)cs_calloc (m, sizeof (csi)) ;              /* get workspace */
        if (!C || !w) return (cs_done (C, w, NULL, 0)) ;     /* out of memory */
        Cp = C->p ; Ci = C->i ; Cx = C->x ;
        for (p = 0 ; p < Ap [n] ; p++) w [Ai [p]]++ ;        /* row counts */
        cs_cumsum (Cp, w, m) ;                              /* row pointers */
        for (j = 0 ; j < n ; j++)
        {
            for (p = Ap [j] ; p < Ap [j+1] ; p++)
            {
                Ci [q = w [Ai [p]]++] = j ;    /* place A(i,j) as entry C(j,i) */
                if (Cx) Cx [q] = Ax [p] ;
            }
        }

        return (cs_done (C, w, NULL, 1)) ;  /* success; free w and return C */
}
```

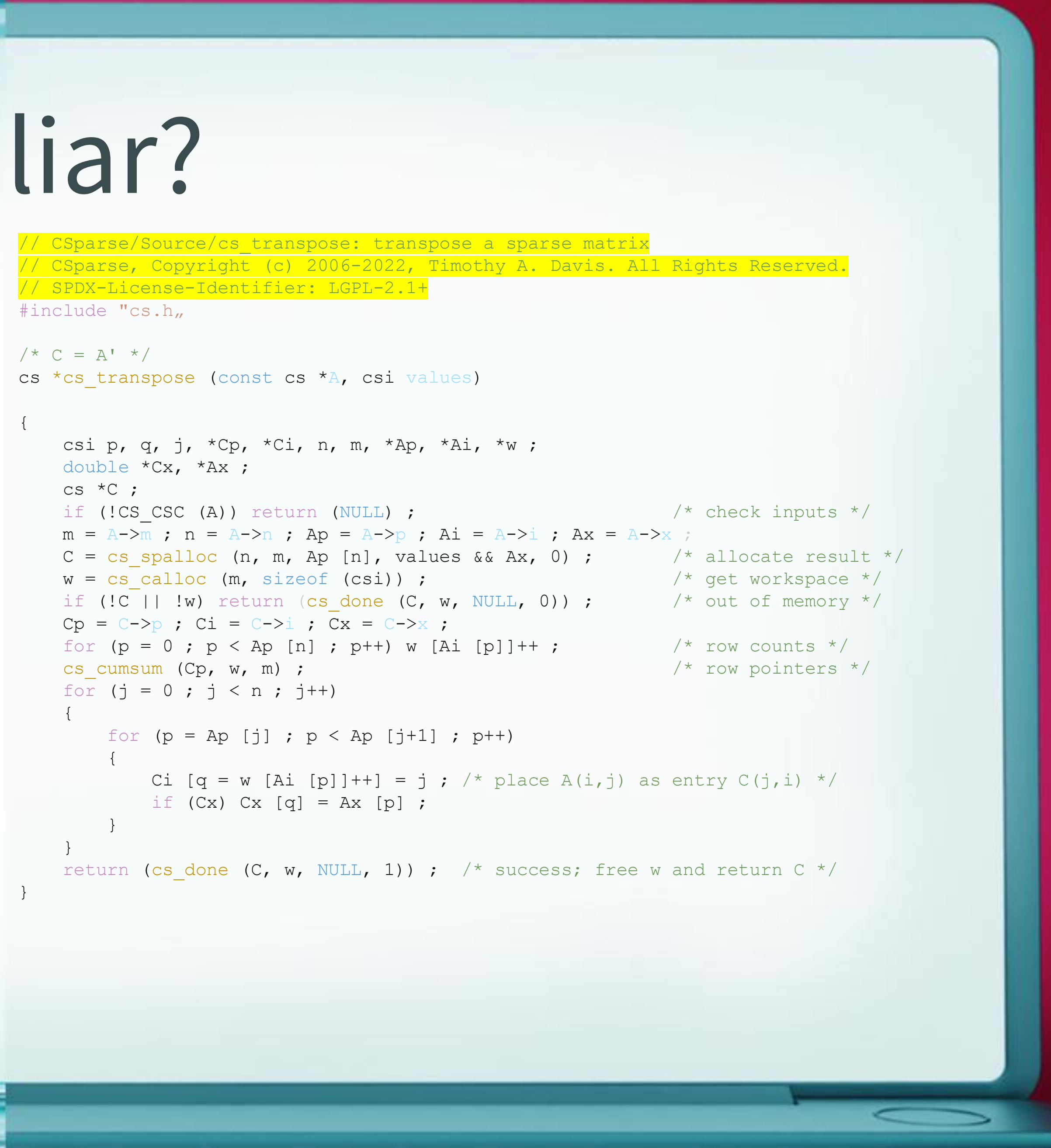


```
// CSparse/Source/cs_transpose: transpose a sparse matrix
// CSparse, Copyright (c) 2006-2022, Timothy A. Davis. All Rights Reserved.
// SPDX-License-Identifier: LGPL-2.1+
#include "cs.h„

/* C = A' */
cs *cs_transpose (const cs *A, csi values)

{
    csi p, q, j, *Cp, *Ci, n, m, *Ap, *Ai, *w ;
    double *Cx, *Ax ;
    cs *C ;
    if (!CS_CSC (A)) return (NULL) ;                     /* check inputs */
    m = A->m ; n = A->n ; Ap = A->p ; Ai = A->i ; Ax = A->x ;
    C = cs_spalloc (n, m, Ap [n], values && Ax, 0) ;      /* allocate result */
    w = cs_calloc (m, sizeof (csi)) ;                     /* get workspace */
    if (!C || !w) return (cs_done (C, w, NULL, 0)) ;      /* out of memory */
    Cp = C->p ; Ci = C->i ; Cx = C->x ;
    for (p = 0 ; p < Ap [n] ; p++) w [Ai [p]]++ ;         /* row counts */
    cs_cumsum (Cp, w, m) ;                               /* row pointers */
    for (j = 0 ; j < n ; j++)
    {
        for (p = Ap [j] ; p < Ap [j+1] ; p++)
        {
            Ci [q = w [Ai [p]]++] = j ; /* place A(i,j) as entry C(j,i) */
            if (Cx) Cx [q] = Ax [p] ;
        }
    }
    return (cs_done (C, w, NULL, 1)) ;  /* success; free w and return C */
}
```

Similiar?



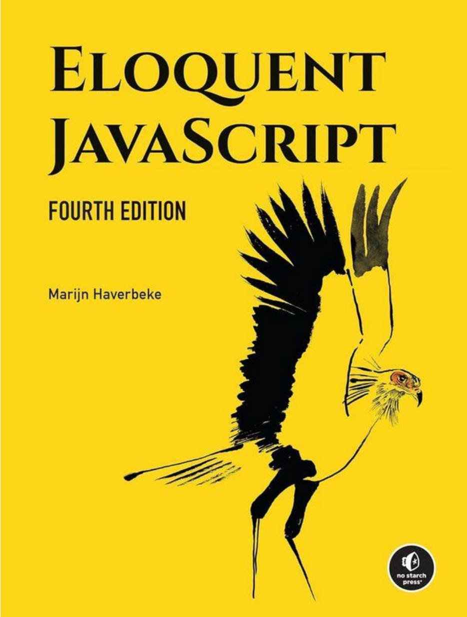
```
"/* C = A' */"
cs *cs_transpose (const cs *A, csi values)


{
     csi p, q, j, *Cp, *Ci, n, m, *Ap, *Ai, *w ;
        double *Cx, *Ax ;
        cs *C ;
        if (!CS_CSC (A)) return (NULL) ;                  /* check inputs */
        m = A->m ; n = A->n ; Ap = A->p ; Ai = A->i ; Ax = A->x ;
        C = cs_spalloc (n, m, Ap [n], values && Ax, 0) ;  /* alloc result */
        w = (csi*)cs_calloc (m, sizeof (csi)) ;           /* get workspace */
        if (!C || !w) return (cs_done (C, w, NULL, 0)) ;  /* out of memory */
        Cp = C->p ; Ci = C->i ; Cx = C->x ;
        for (p = 0 ; p < Ap [n] ; p++) w [Ai [p]]++ ;     /* row counts */
        cs_cumsum (Cp, w, m) ;                            /* row pointers */
        for (j = 0 ; j < n ; j++)
        {
            for (p = Ap [j] ; p < Ap [j+1] ; p++)
            {
                Ci [q = w [Ai [p]]++] = j ;    /* place A(i,j) as entry C(j,i) */
                if (Cx) Cx [q] = Ax [p] ;
            }
        }

        return (cs_done (C, w, NULL, 1)) ;  /* success; free w and return C */
}
```



```
// CSparse/Source/cs_transpose: transpose a sparse matrix
// CSparse, Copyright (c) 2006-2022, Timothy A. Davis. All Rights Reserved.
// SPDX-License-Identifier: LGPL-2.1+
#include "cs.h„

/* C = A' */
cs *cs_transpose (const cs *A, csi values)

{
    csi p, q, j, *Cp, *Ci, n, m, *Ap, *Ai, *w ;
    double *Cx, *Ax ;
    cs *C ;
    if (!CS_CSC (A)) return (NULL) ;                    /* check inputs */
    m = A->m ; n = A->n ; Ap = A->p ; Ai = A->i ; Ax = A->x ;
    C = cs_spalloc (n, m, Ap [n], values && Ax, 0) ;    /* allocate result */
    w = cs_calloc (m, sizeof (csi)) ;                   /* get workspace */
    if (!C || !w) return (cs_done (C, w, NULL, 0)) ;    /* out of memory */
    Cp = C->p ; Ci = C->i ; Cx = C->x ;
    for (p = 0 ; p < Ap [n] ; p++) w [Ai [p]]++ ;       /* row counts */
    cs_cumsum (Cp, w, m) ;                              /* row pointers */
    for (j = 0 ; j < n ; j++)
    {
        for (p = Ap [j] ; p < Ap [j+1] ; p++)
        {
            Ci [q = w [Ai [p]]++] = j ; /* place A(i,j) as entry C(j,i) */
            if (Cx) Cx [q] = Ax [p] ;
        }
    }
    return (cs_done (C, w, NULL, 1)) ;  /* success; free w and return C */
}
```

KI generated “IsEven”

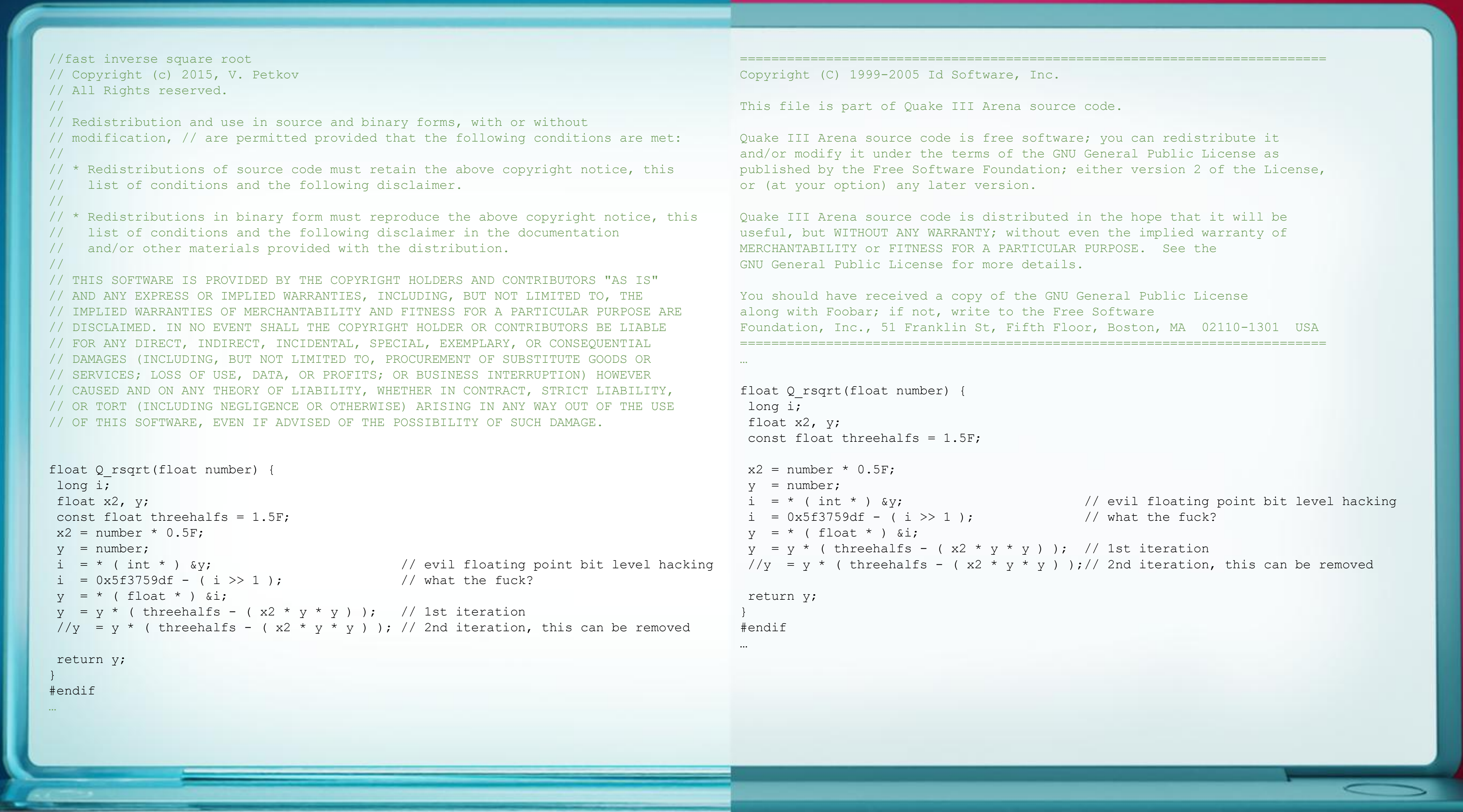
```
function isEven(n)


{
  if (n == 0)
    return true;
  else if (n == 1)
    return false;
  else if (n < 0)
    return isEven(-n);
  else
    return isEven(n - 2);
}

console.log(isEven(50));
// → true
console.log(isEven(75));
// → false
console.log(isEven(-1));
// → ??
```



Eloquent Javascript by Marijn Haverbeke.
https://eloquentjavascript.net/code/#3.2
Buch CC-BY-NC, Code MIT

Experienced Developers

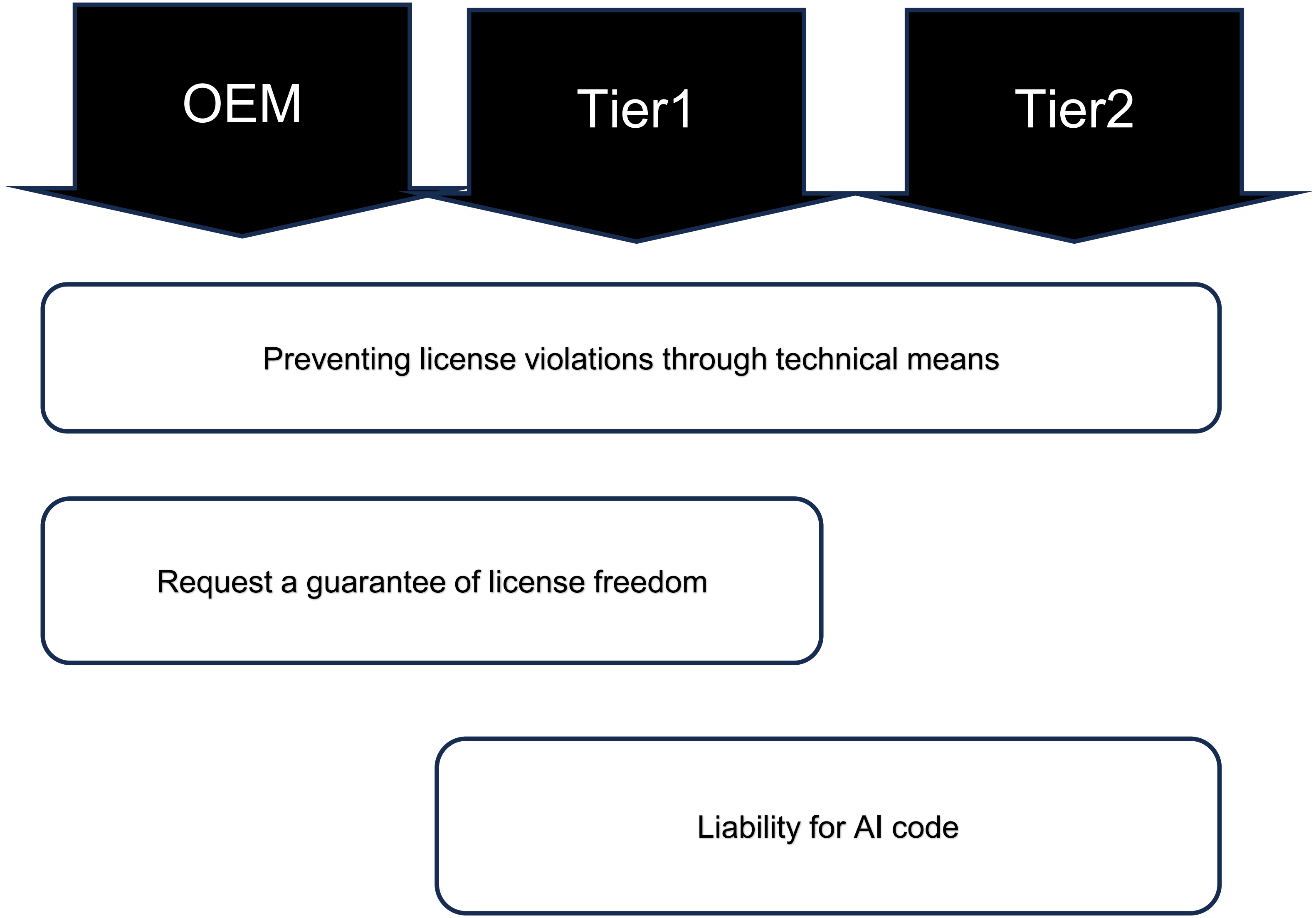
```
function isEven(n)


{
return n % 2 === 0;
}
```



```
//fast inverse square root
// Copyright (c) 2015, V. Petkov
// All Rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, // are permitted provided that the following conditions are met:
//
// * Redistributions of source code must retain the above copyright notice, this
//   list of conditions and the following disclaimer.
//
// * Redistributions in binary form must reproduce the above copyright notice, this
//   list of conditions and the following disclaimer in the documentation
//   and/or other materials provided with the distribution.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
// AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
// IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
// DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
// FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
// DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
// SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
// CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
// OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.


float Q_rsqrt(float number) {
 long i;
 float x2, y;
 const float threehalfs = 1.5F;
 x2 = number * 0.5F;
 y  = number;
 i  = * ( int * ) &y;                        // evil floating point bit level hacking
 i  = 0x5f3759df - ( i >> 1 );               // what the fuck?
 y  = * ( float * ) &i;
 y  = y * ( threehalfs - ( x2 * y * y ) );   // 1st iteration
 //y  = y * ( threehalfs - ( x2 * y * y ) ); // 2nd iteration, this can be removed

 return y;
}
#endif
…
```

```
===========================================================================
Copyright (C) 1999-2005 Id Software, Inc.

This file is part of Quake III Arena source code.

Quake III Arena source code is free software; you can redistribute it
and/or modify it under the terms of the GNU General Public License as
published by the Free Software Foundation; either version 2 of the License,
or (at your option) any later version.

Quake III Arena source code is distributed in the hope that it will be
useful, but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with Foobar; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA  02110-1301  USA
===========================================================================
…

float Q_rsqrt(float number) {
 long i;
 float x2, y;
 const float threehalfs = 1.5F;

 x2 = number * 0.5F;
 y  = number;
 i  = * ( int * ) &y;                       // evil floating point bit level hacking
 i  = 0x5f3759df - ( i >> 1 );              // what the fuck?
 y  = * ( float * ) &i;
 y  = y * ( threehalfs - ( x2 * y * y ) );  // 1st iteration
 //y  = y * ( threehalfs - ( x2 * y * y ) );// 2nd iteration, this can be removed

 return y;
}
#endif
…
```

Solve the problem yourself or have it solved



OEM

Tier1

Tier2

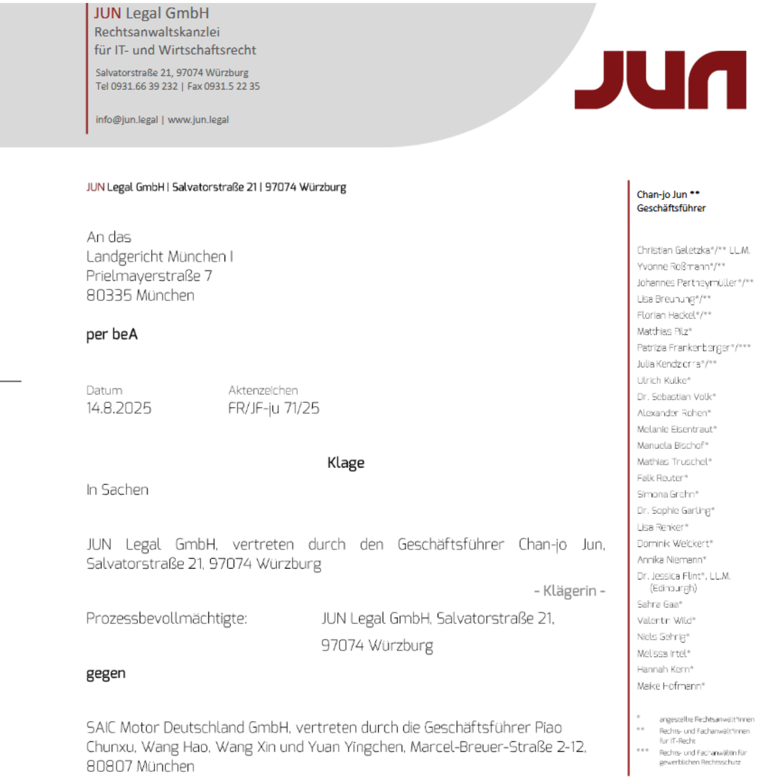
Preventing license violations through technical means

Request a guarantee of license freedom

Liability for AI code

Usage(Good faith)=/dev/zero

- Infringement regardless of fault
- Munich Regional Court Jun./. SAIC: Warranty claim due to FOSS infringement



JUN Legal GmbH
Rechtsanwaltskanzlei
für IT- und Wirtschaftsrecht
Salvatorstraße 21, 97074 Würzburg
Tel 0931.66 39 232 | Fax 0931.5 22 35
info@jun.legal | www.jun.legal

JUN

JUN Legal GmbH | Salvatorstraße 21 | 97074 Würzburg

An das
Landgericht München I
Prielmayerstraße 7
80335 München

per beA

Datum	Aktenzeichen
14.8.2025	FR/JF-ju 71/25

Klage

In Sachen

JUN Legal GmbH, vertreten durch den Geschäftsführer Chan-jo Jun, Salvatorstraße 21, 97074 Würzburg

- Klägerin -

Prozessbevollmächtigte: JUN Legal GmbH, Salvatorstraße 21,
97074 Würzburg

gegen

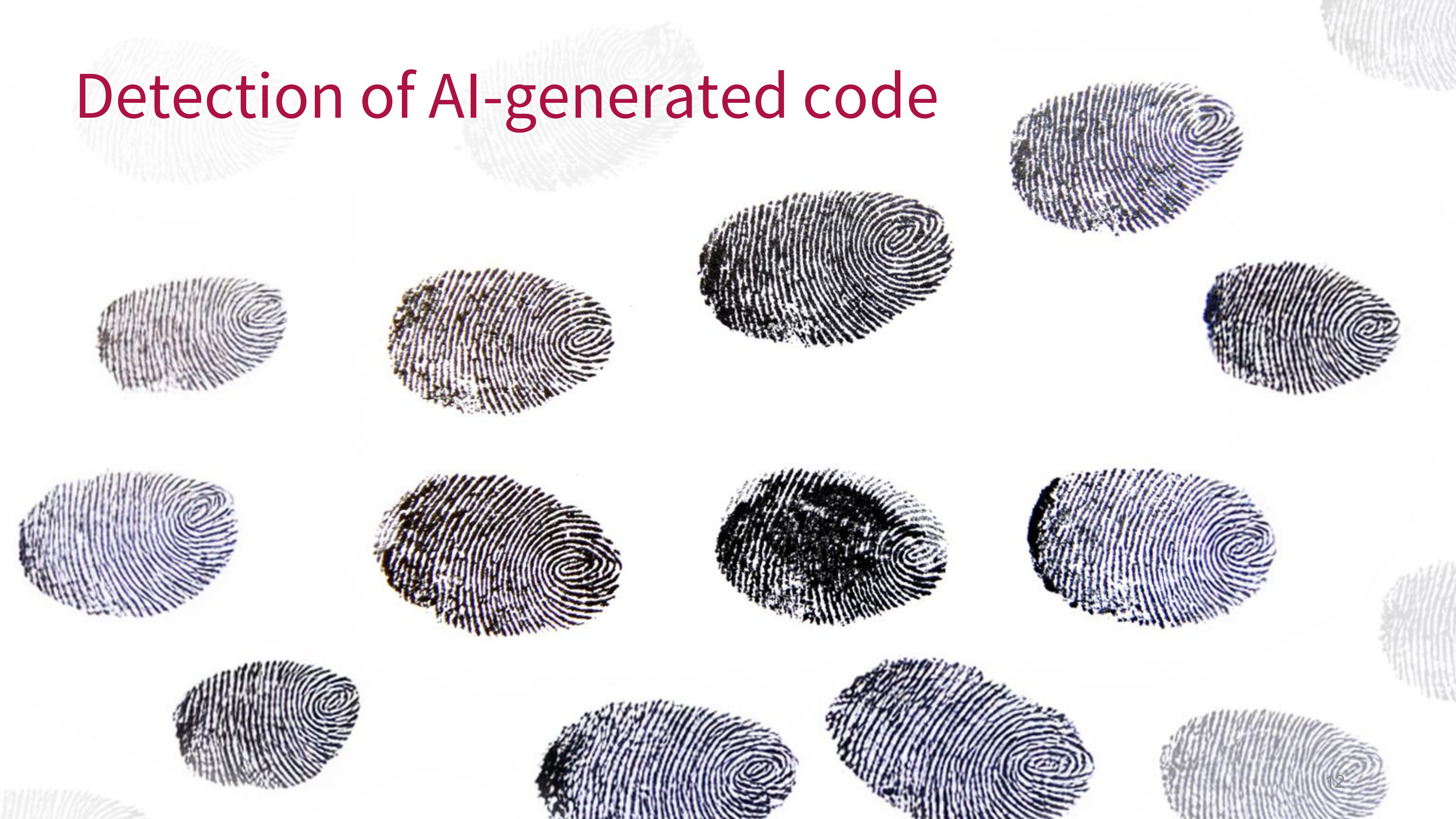
SAIC Motor Deutschland GmbH, vertreten durch die Geschäftsführer Piao Chunxu, Wang Hao, Wang Xin und Yuan Yingchen, Marcel-Breuer-Straße 2-12, 80807 München

Chan-jo Jun **
Geschäftsführer

Christian Geletzka*/** LL.M.
Yvonne Roßmann*/**
Johannes Partheymüller*/**
Lisa Breunung*/**
Florian Hackel*/**
Matthias Pilz*
Patrizia Frankenberger*/***
Julia Kendziorra*/**
Ulrich Kulke*
Dr. Sebastian Volk*
Alexander Rohen*
Melanie Eisentraut*
Manuela Bischof*
Mathias Truschel*
Falk Reuter*
Simona Grehn*
Dr. Sophie Garling*
Lisa Renker*
Dominik Weickert*
Annika Niemann*
Dr. Jessica Flint*, LL.M. (Edinburgh)
Sahra Gaa*
Valentin Wild*
Niels Gehrig*
Melissa Irtel*
Hannah Kern*
Maike Hofmann*

* angestellte Rechtsanwält*innen
** Rechts- und Fachanwält*innen für IT-Recht
*** Rechts- und Fachanwältin für gewerblichen Rechtsschutz

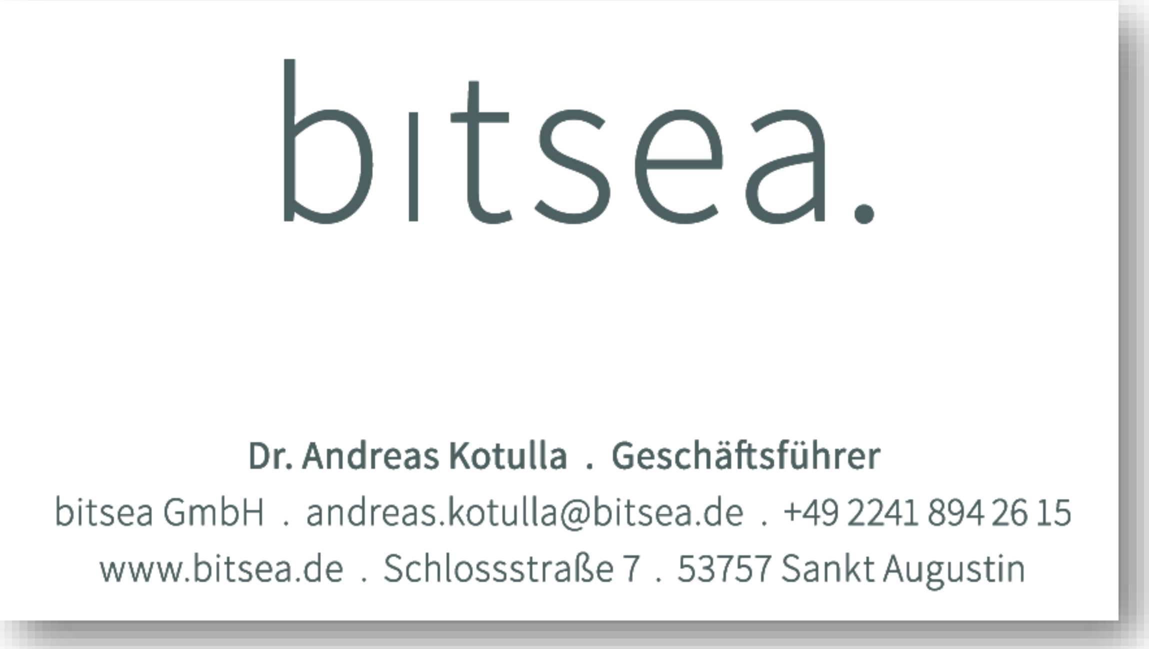
Detection of AI-generated code



Protective Measures: What Developers and Companies Need to Do Now

Summary

Contact

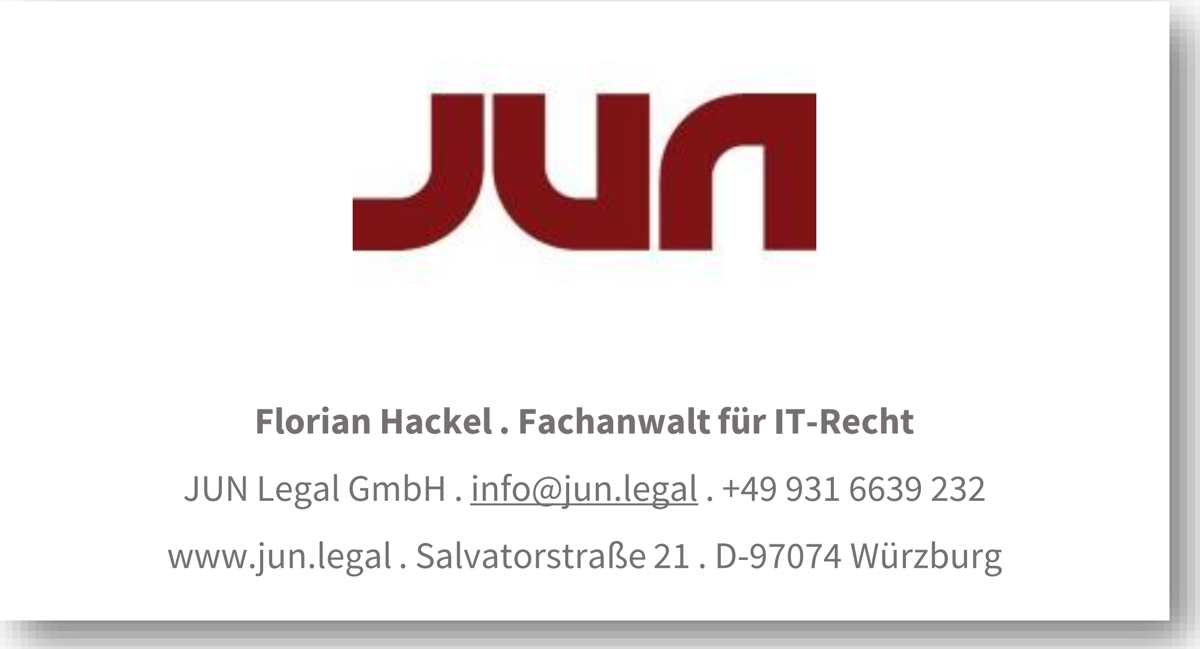


bitsea.

Dr. Andreas Kotulla . Geschäftsführer

bitsea GmbH . andreas.kotulla@bitsea.de . +49 2241 894 26 15

www.bitsea.de . Schlossstraße 7 . 53757 Sankt Augustin



Florian Hackel . Fachanwalt für IT-Recht

JUN Legal GmbH . info@jun.legal . +49 931 6639 232

www.jun.legal . Salvatorstraße 21 . D-97074 Würzburg