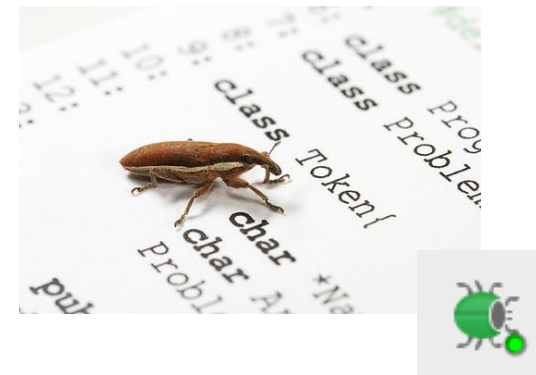


Debugging

Help with finding bugs in your code

Produced by: Dr. Siobhán Drohan
Ms. Mairéad Meagher
Siobhan Roche

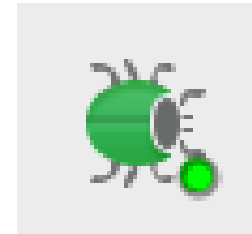


Topic List

1. What are **bugs**?



2. What are **debuggers**?



3. How do I use them?

A small, brown, elongated beetle with a lighter-colored underbelly is standing on a white surface. The surface is covered with printed C++ code, including lines like 'class Token{', 'char *Na', 'char Probl', and 'pub'. The beetle is positioned in the center-left of the frame, facing right. The background is slightly blurred, showing more of the code and some green text at the top right.

1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
26

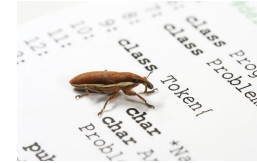
https://en.wikipedia.org/wiki/Software_bug

Bugs can be frustrating to find/fix

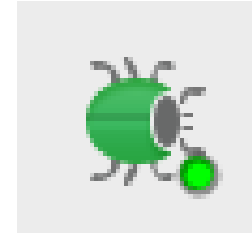


Topic List

1. What are **bugs**?



2. What are **debuggers**?



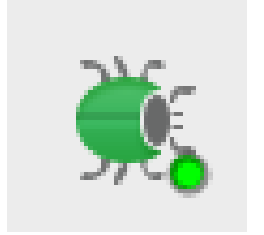
3. How do I use them?

Help is at hand...debuggers!

A debugger can be **used to fix bugs**

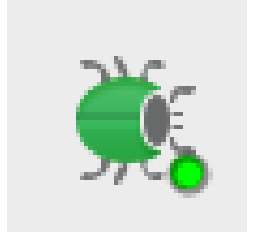
...hence the name debugger!

Debugger



- A **debugger** is a software tool that
 - helps in examining how an application executes
 - allows programmers execute an application one statement at a time.
 - **Step, step into, step out.**
 - typically provides functions
 - to stop and start a program at selected points in the source code(**breakpoints**)
 - to examine the values of variables (**watch, trace**)

Debugger



- Debuggers are especially useful when your program contains **logical errors**.
 - i.e. errors that the compiler will not pickup but that lead to incorrect results
 - e.g. if your syntax is correct but the logic of your problem solution is faulty.
- Using the debugger, you can **trace** how each of the calculations and changes made to fields/variables happen and hopefully **figure out where the error is occurring**.

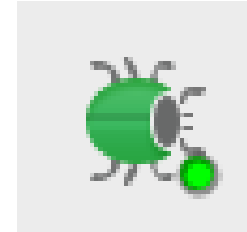


Topic List

1. What are **bugs**?

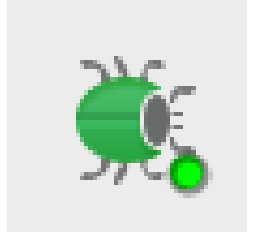


2. What are **debuggers**?



3. How do I use them?

Debugger



- Most IDEs come with a debugger; **IntelliJ** has one.
- We are going to use the **IntelliJ Debugger** to **step** through the debugging of a small program
 - The program iterates over a primitive array of *int* and prints out the largest number in the array.

```
public class Driver {  
  
    public static void main(String args[])  
    {  
        int list[] = {2,5,3,4};  
        int largestNumber = Largest.findLargest(list);  
        System.out.println("Largest number is: " + largestNumber);  
    }  
}
```

Given this code...

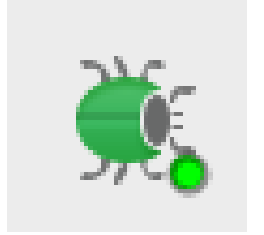
We are expecting this output:

Largest number is: 5

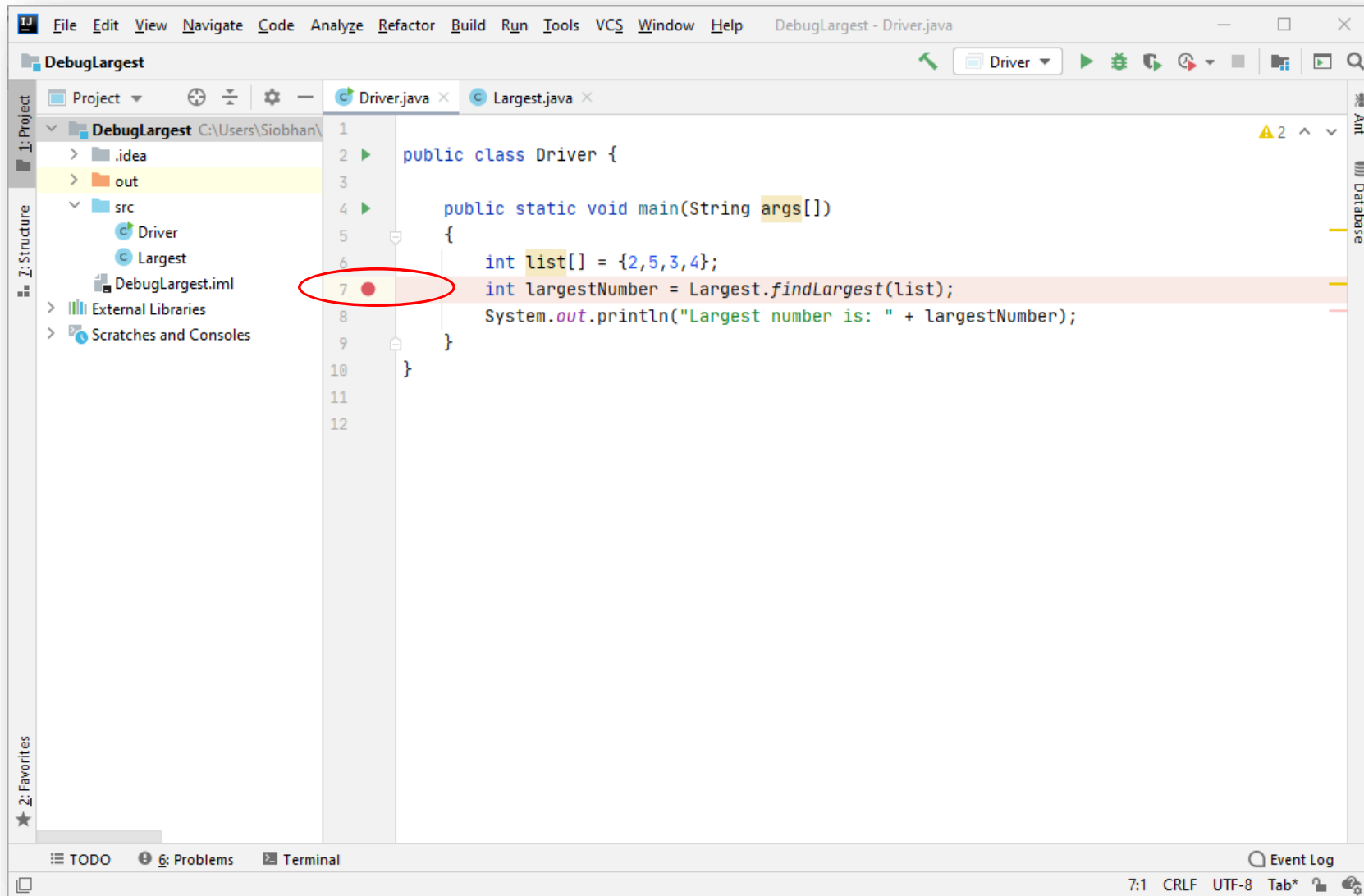
But we get:

Largest number is: 2147483647

```
public class Largest {  
  
    public static int findLargest (int[] list) {  
        int index = 0;  
        int max = Integer.MAX_VALUE;  
  
        for (index = 0; index < list.length; index++) {  
            if (list[index] > max) {  
                max = list[index];  
            }  
        }  
  
        return max;  
    }  
}
```

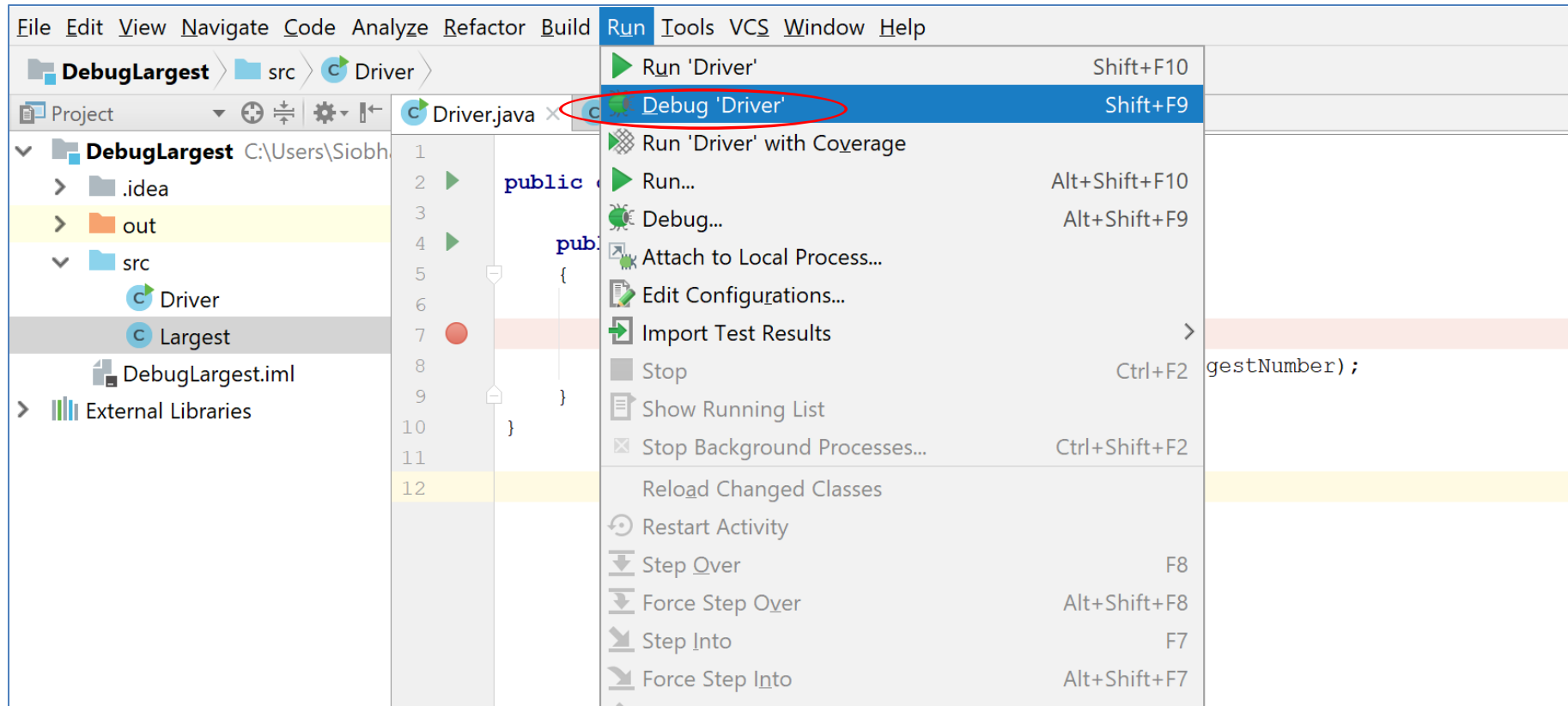


Let's debug the code
in IntelliJ
to help us find the error...



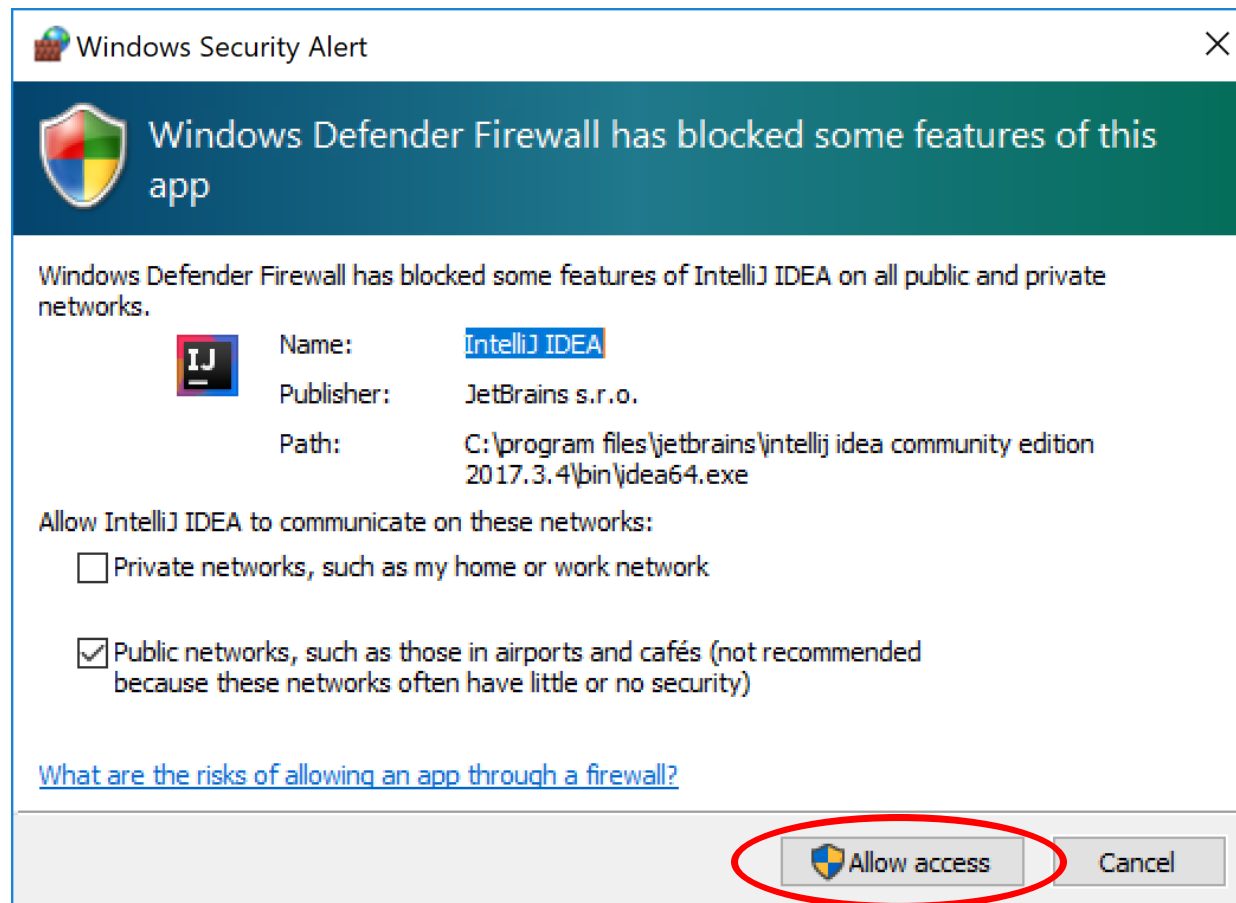
1

Click in the grey margin beside line 7. This will set up a **breakpoint** on this line.



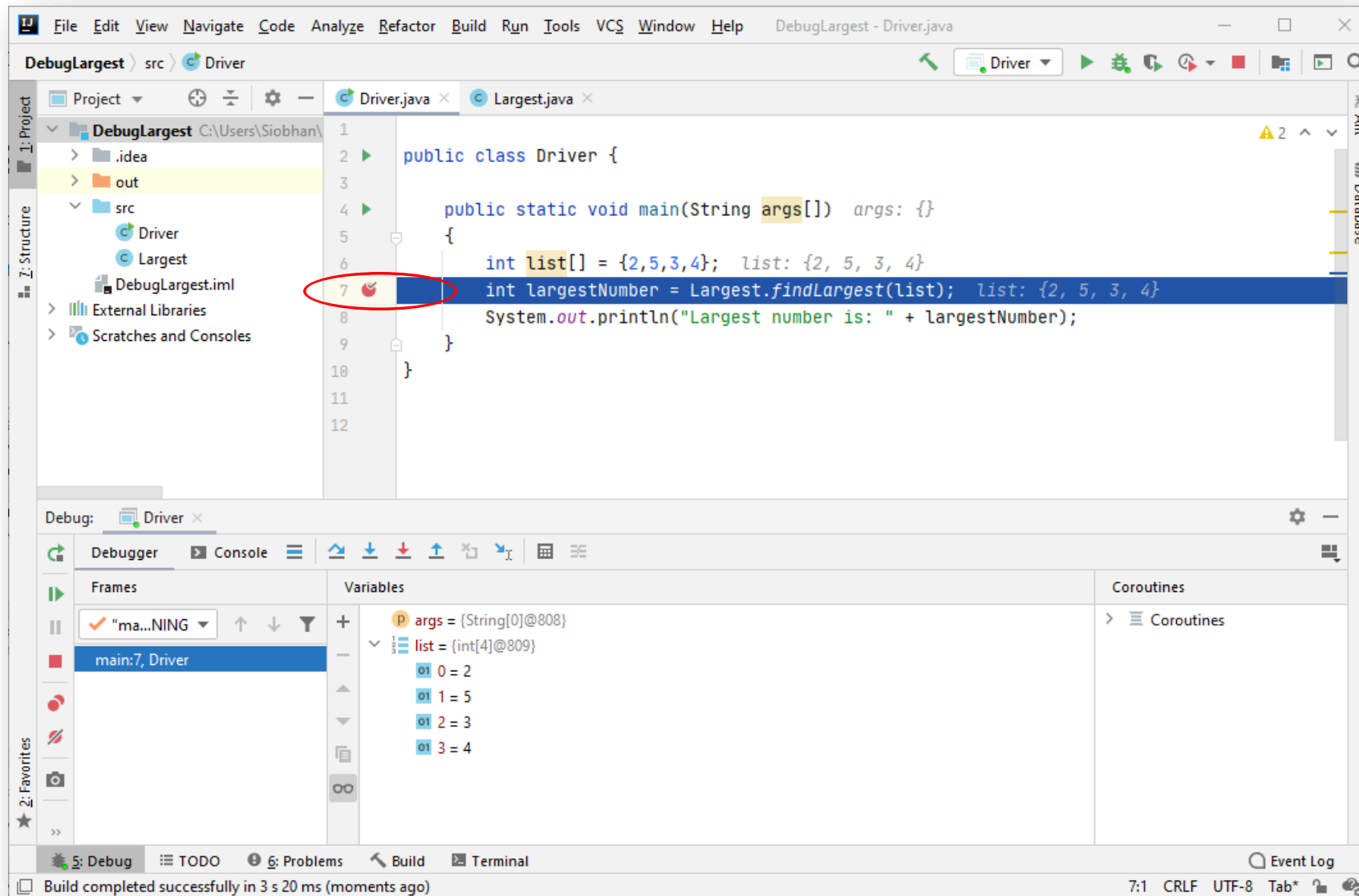
2

From the Run menu, select **Debug** 'Driver'.



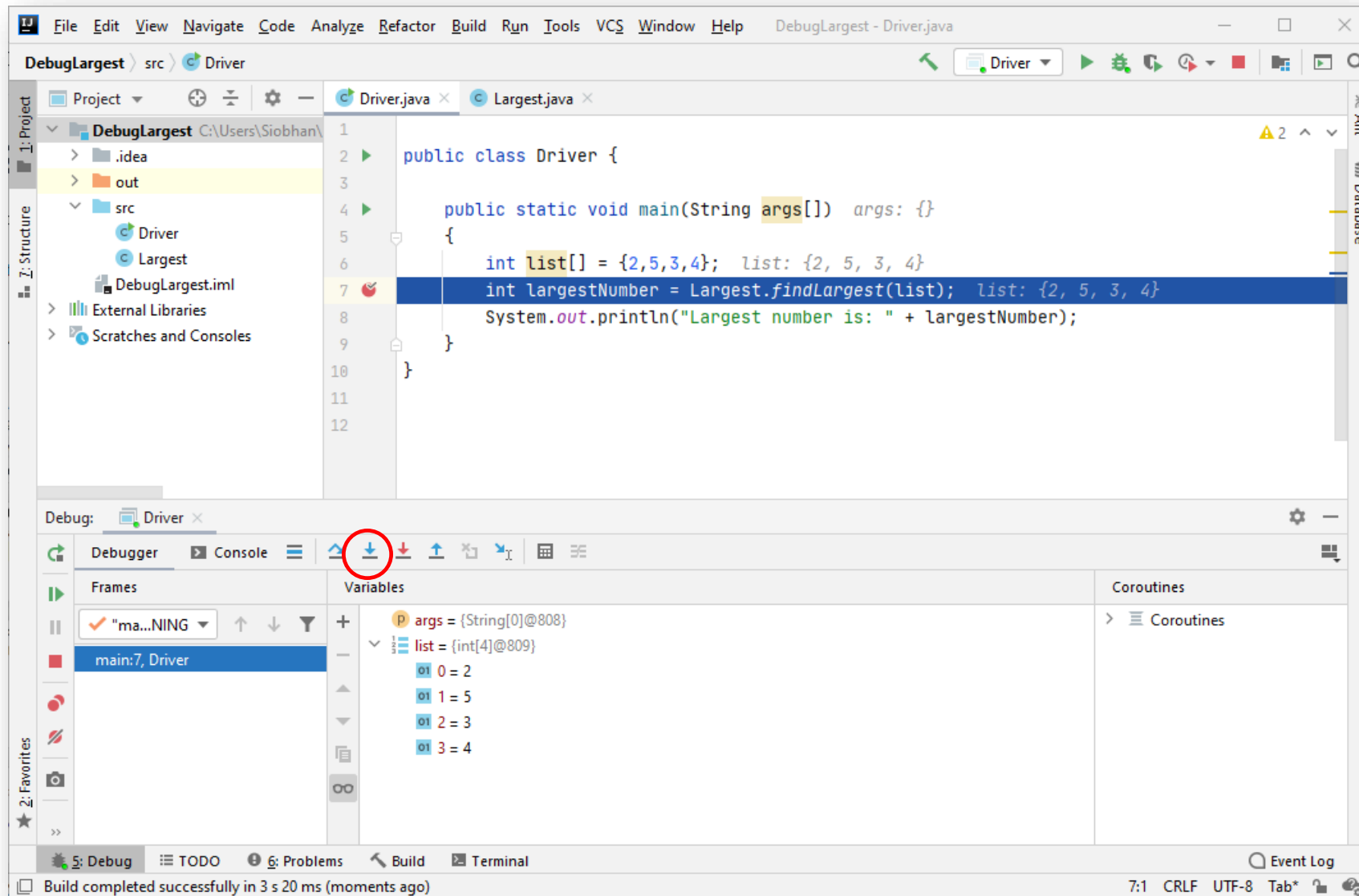
3

If this window appears, click on “Allow access”.



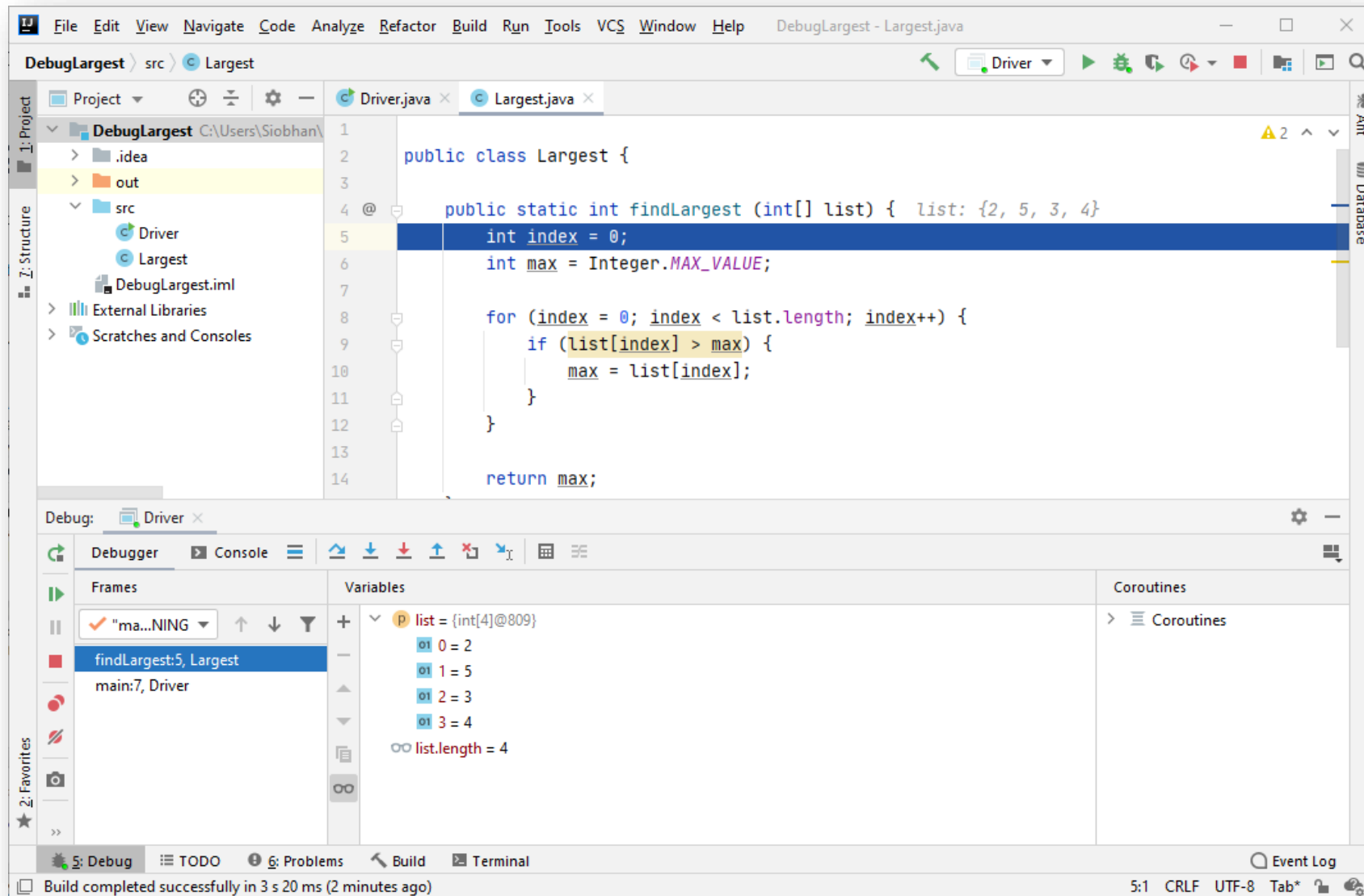
4

You are now in Debug mode...the program has stopped just before executing line 7.



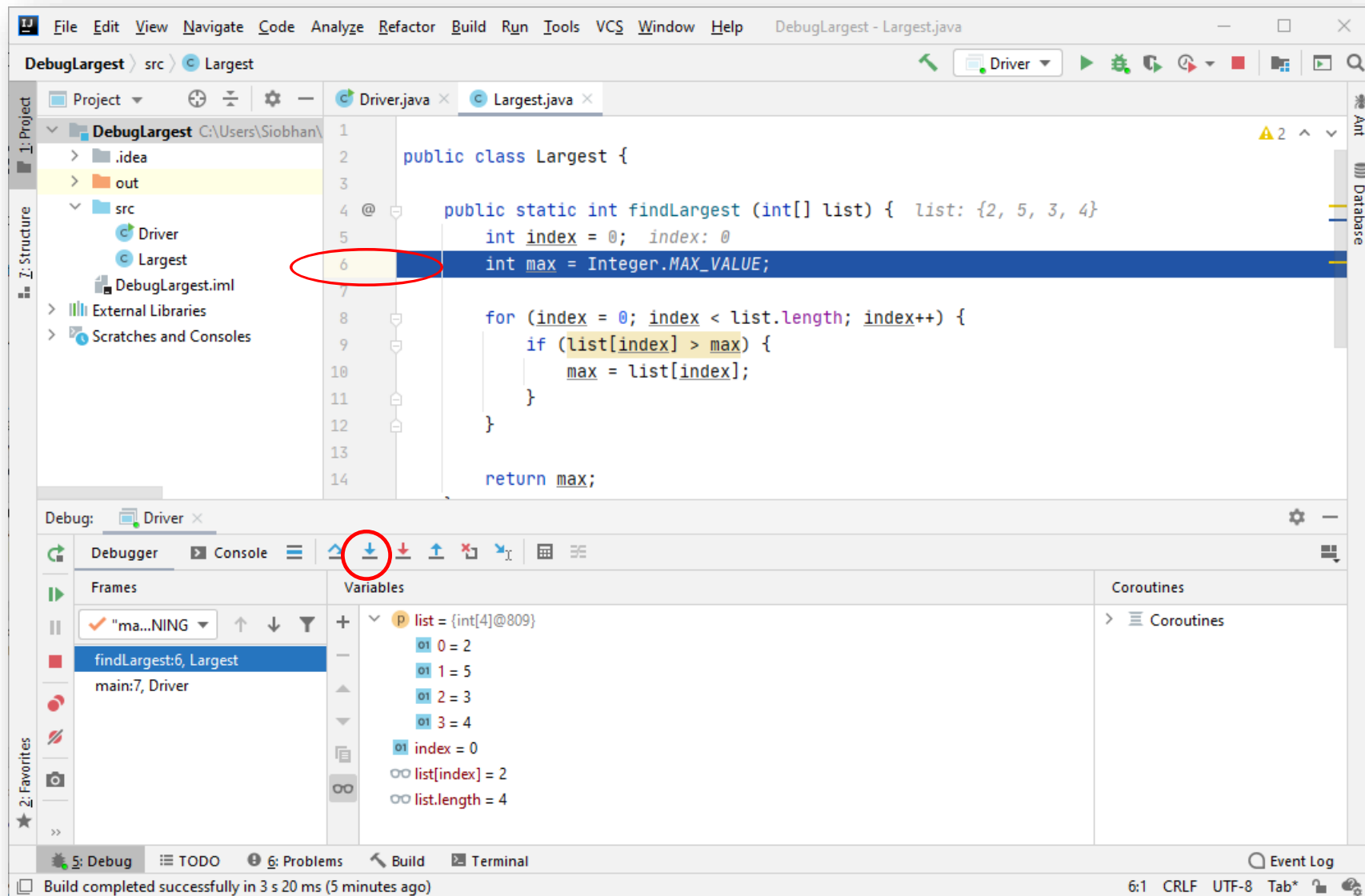
5

'Step Into' the findLargest method...



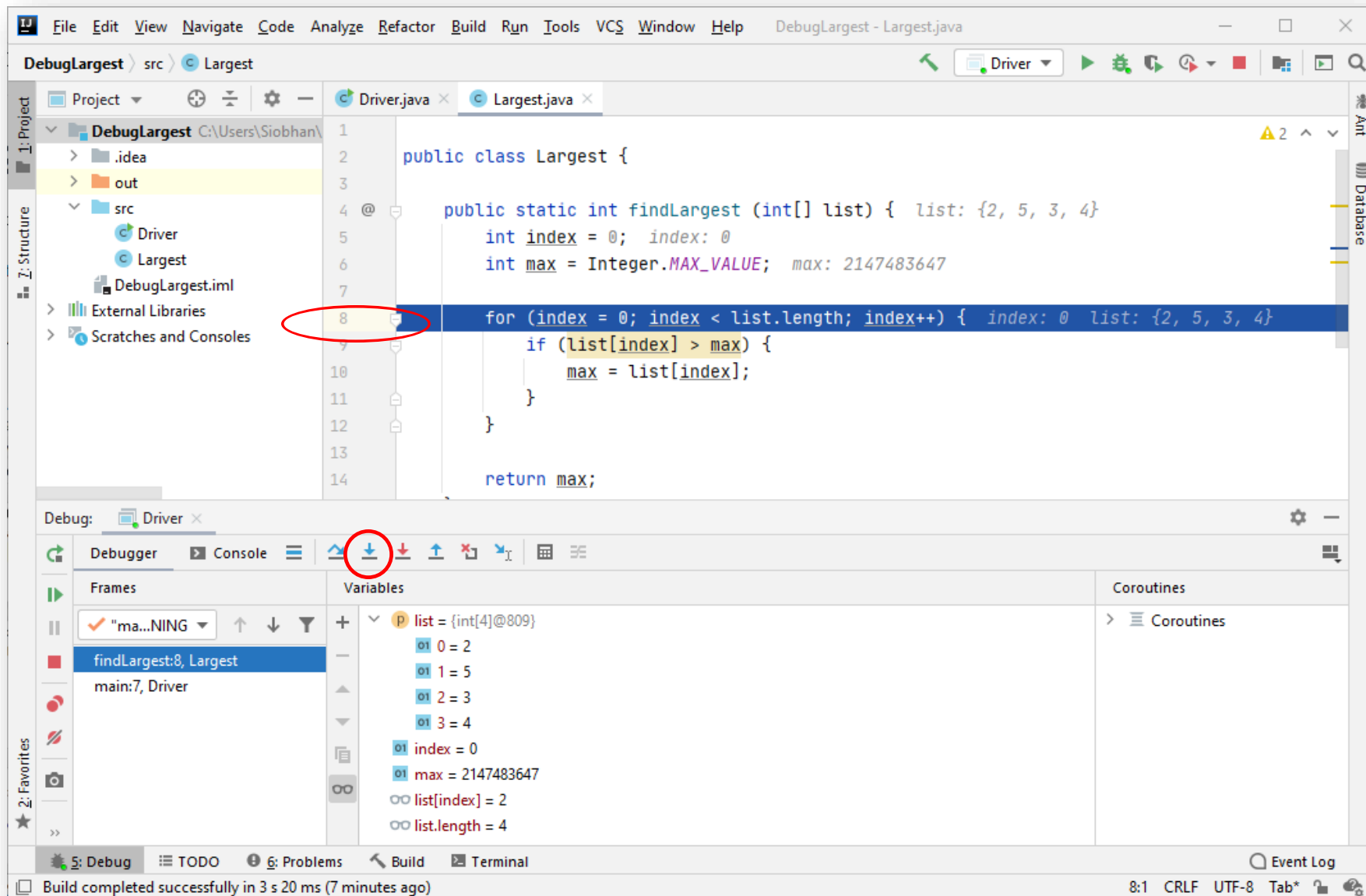
6

Now that we are in the `findLargest` method, we want to execute each line while monitoring the value of `Max`. This should help us locate the error...



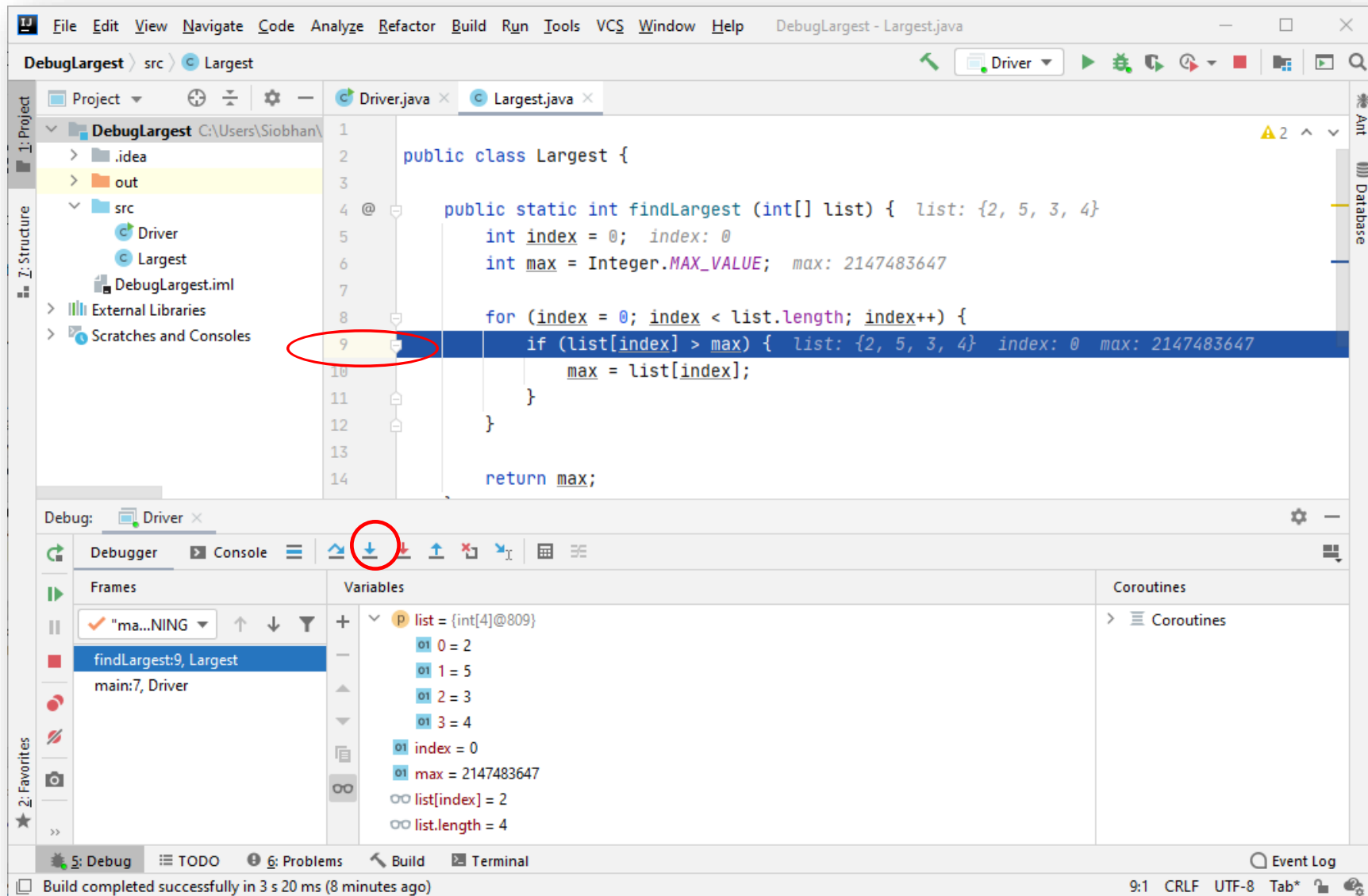
7

'Step Into' again → execution now stopped on line 6



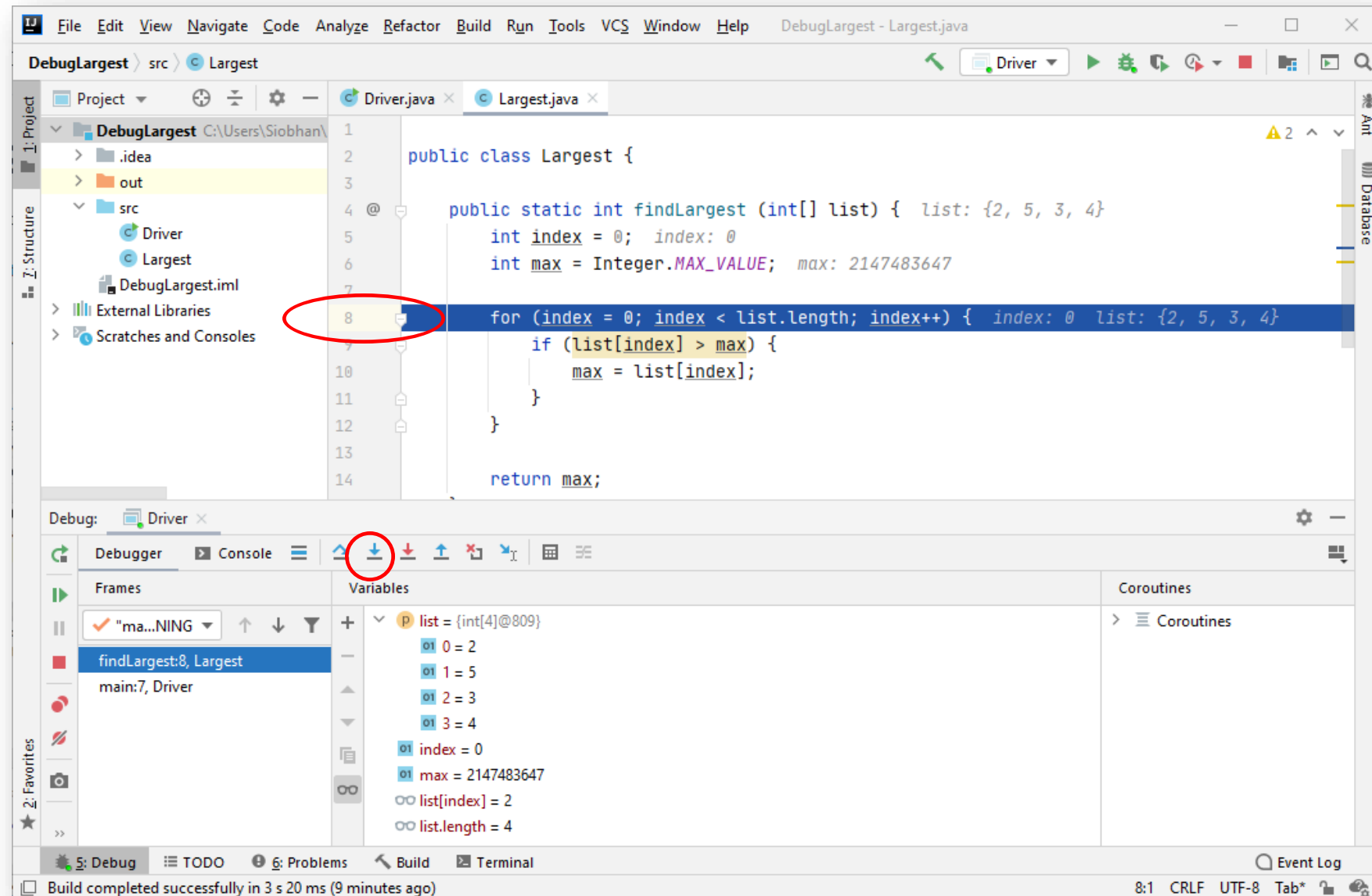
8

Step Into → execution now stopped on line 8...note the value of **max**.



9

Step Into → execution now stopped on line 9...



10

Step Into → execution now stopped back on line 8...
can you see the problem?

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help DebugLargest - Largest.java

DebugLargest src Largest

Project DebugLargest C:\Users\Siobhan\src\out

src Driver Largest DebugLargest.iml External Libraries Scratches and Consoles

```
1 public class Largest {
2
3
4 @ public static int findLargest (int[] list) { list: {2, 5, 3, 4}
5     int index = 0; index: 0
6     int max = Integer.MAX_VALUE; max: 2147483647
7
8     for (index = 0; index < list.length; index++) { index: 0 list: {2, 5, 3, 4}
9         if (list[index] > max) {
10             max = list[index];
11         }
12     }
13
14     return max;
15 }
```

Debug: Driver

Debugger Console

Frames "ma...NING" findLargest:8, Largest main:7, Driver

Variables

- list = {int[4]@809}
- 0 = 2
- 1 = 5
- 2 = 3
- 3 = 4
- index = 0
- max = 2147483647
- list[index] = 2
- list.length = 4

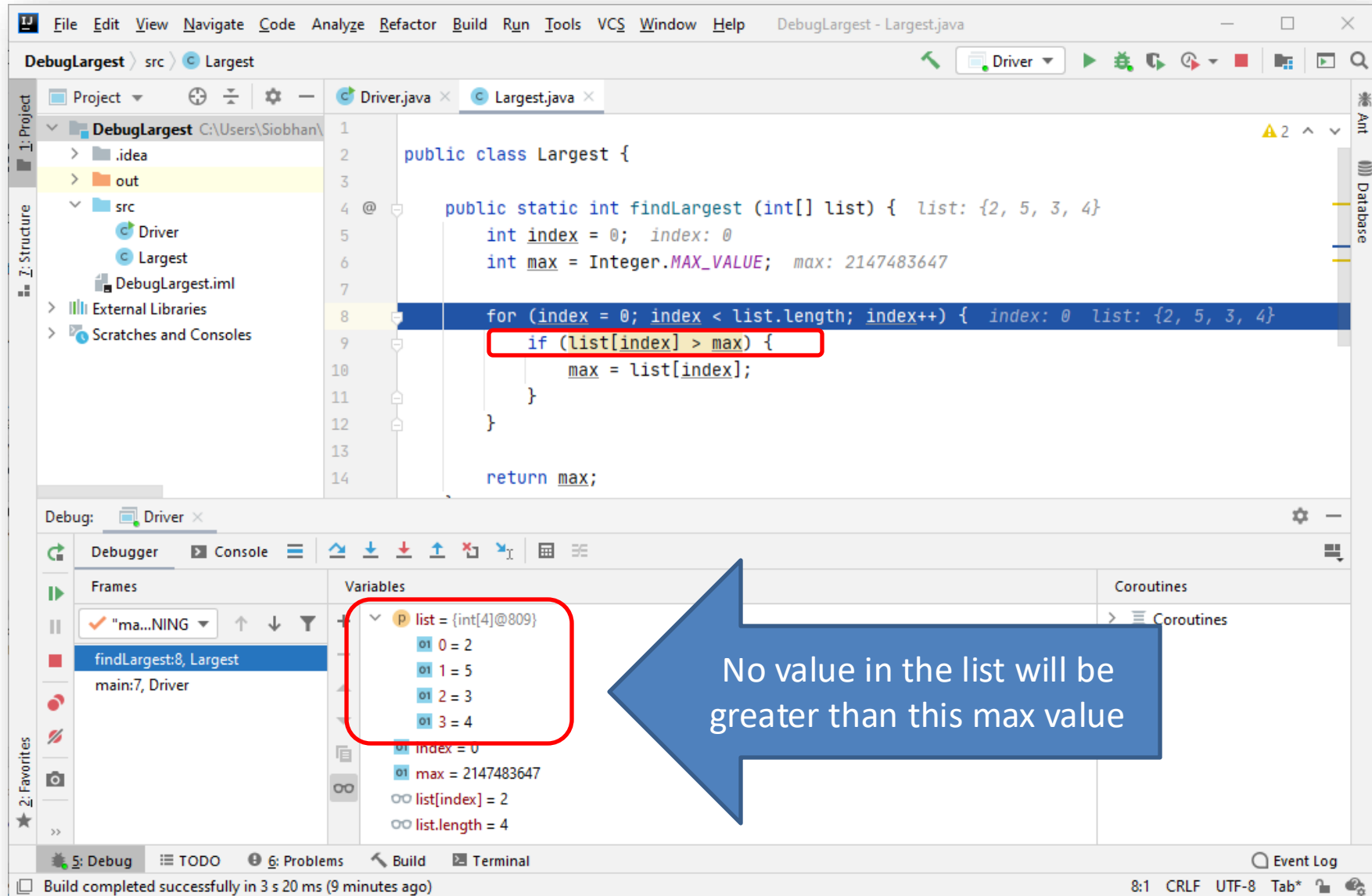
Coroutines

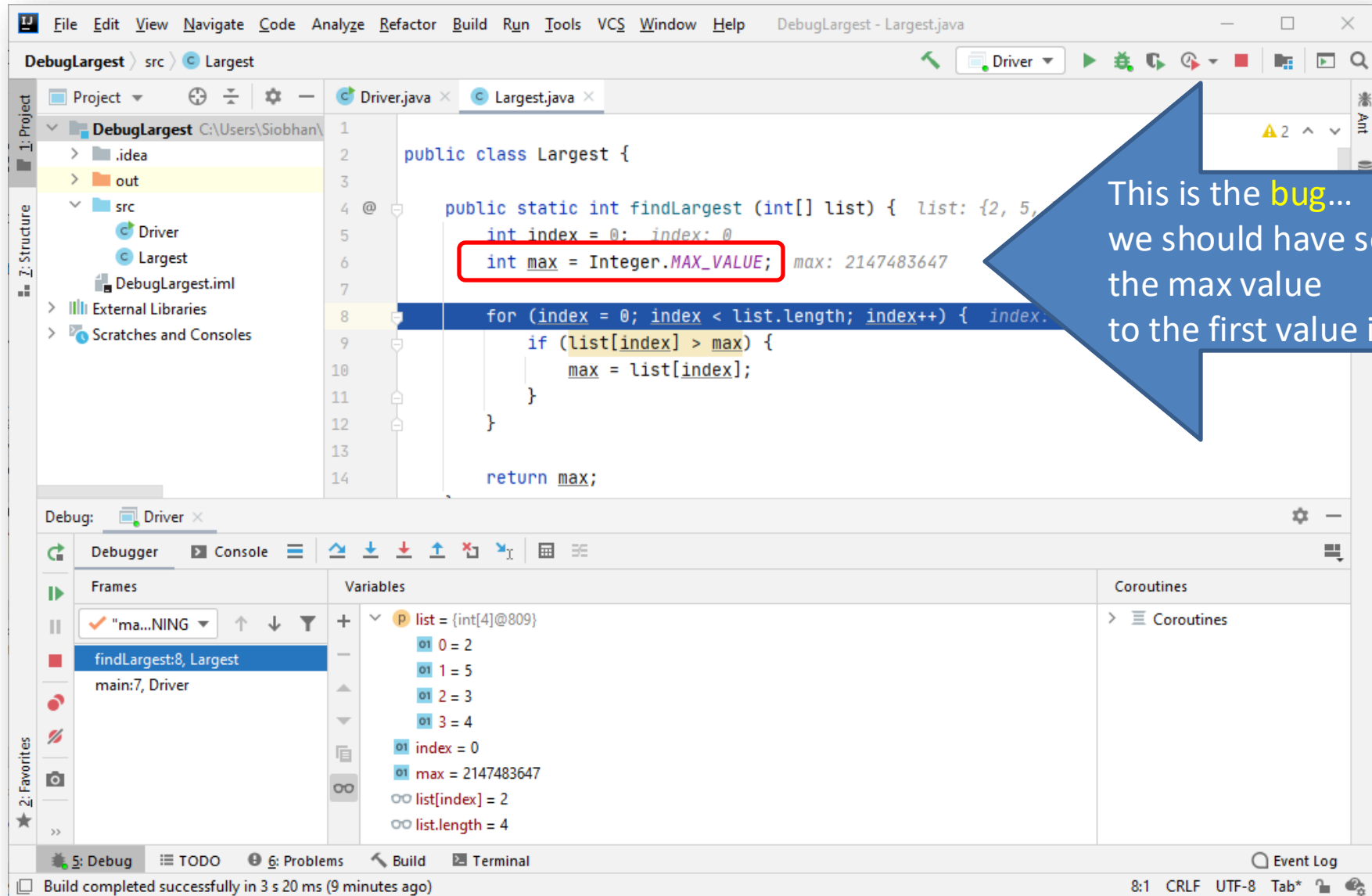
Coroutines

Build completed successfully in 3 s 20 ms (9 minutes ago)

8:1 CRLF UTF-8 Tab*

Event Log





Fixing the bug

- Instead of the line of code:

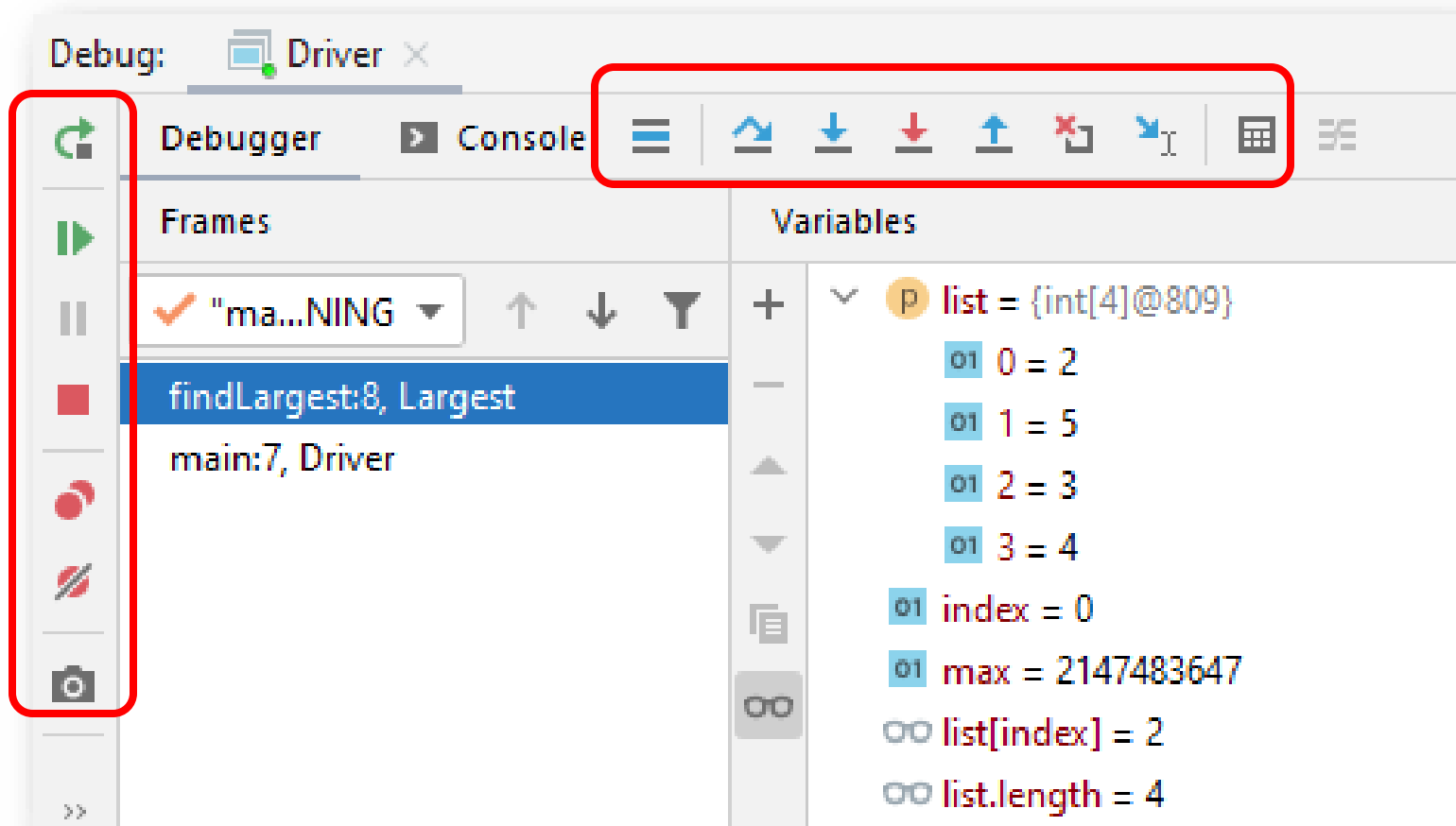
```
int max = Integer.MAX_VALUE;
```

- We need:

```
int max = list[0];
```



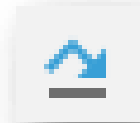
Some IntelliJ debugger buttons...



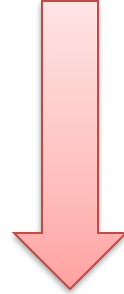
Some Debugger buttons...



- **Step Over**



to step over the next method call (without entering it) at the currently executing line of code.



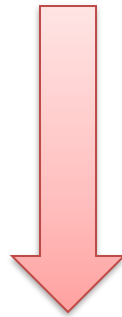
Some Debugger buttons...



- **Step Into**



step into the next method call at the currently executing line of code.



Some Debugger buttons...

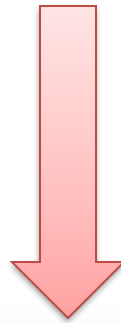


- **Step Out**

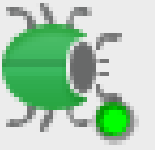


executes the remaining lines of a method in which the current execution point lies.

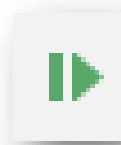
The next statement displayed is the statement following the method call.



Some Debugger buttons...



- **Resume Program**



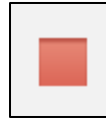
resume the execution of the currently suspended program; it will stop again at the next available breakpoint (if any).



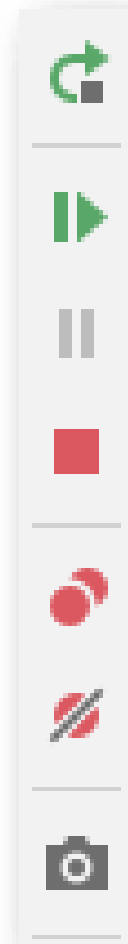
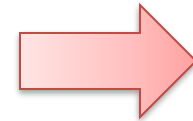
Some Debugger buttons...



- **Terminate**



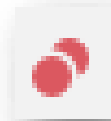
to terminate the launch associated with the selected debug target i.e. **stop** the program.



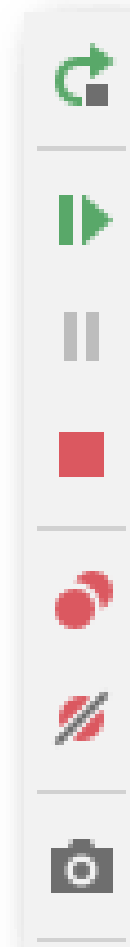
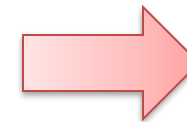
Some Debugger buttons...



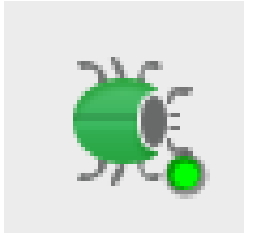
- **Show breakpoints**



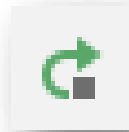
show all the breakpoints (in a pop up window) in the program.



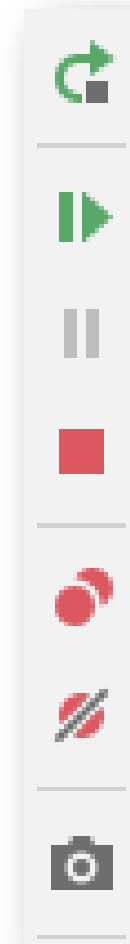
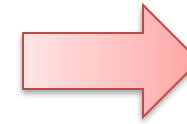
Some Debugger buttons...



- **Rerun program**



start the program again.





Now it's your turn!

- practice using the debugger on this code (there is a step in your labs to provide support).

**Any
Questions?**

