

Домашнє завдання №25_1

Застосовуючи парадигму функційного програмування скласти програму мовою Haskell, яка виконує імплементацію швидкого сортування без використання готових бібліотечних реалізацій.

** зверніть увагу, що блоки коду в Haskell виділяються відступами, що задають так званий двомірний синтаксис, тому будьте уважні при редагуванні наведеного далі прикладу коду і краще не змішуйте пробіли з табуляцією, яка трактується Haskell як 8 пробілів, а текстовими редакторами відображується як 4 пробіли.*

** коментар: це завдання тотожне до завдання №25_2, №26_1, №26_2 та лабораторної роботи №5; таким чином можна порівняти різні засоби програмування; далі наводиться приклад повністю виконаного завдання; для компіляції і запуску коду можна використати <https://repl.it/languages/haskell> .*

Вибір варіанту

$$(N_{\text{ж}} + N_{\text{г}} + 1) \% 30 + 1$$

де: $N_{\text{ж}}$ – порядковий номер студента в групі, а $N_{\text{г}}$ – номер групи (1,2,3,4,5,6,7,8 або 9)

Варіанти завдань

Варіант	Вхідні дані
1	125, 126, 121, 121, 117, 120, 121, 116, 116, 112, 119, 111, 116, 117, 112, 114, 106, 111, 112, 107
2	145, 146, 141, 141, 137, 140, 141, 136, 136, 132, 139, 131, 136, 137, 132, 134, 126, 131, 132, 12
3	165, 166, 161, 161, 157, 160, 161, 156, 156, 152, 159, 151, 156, 157, 152, 154, 146, 151, 152, 147
4	185, 186, 181, 181, 177, 180, 181, 176, 176, 172, 179, 171, 176, 177, 172, 174, 166, 171, 172, 167
5	205, 206, 201, 201, 197, 200, 201, 196, 196, 192, 199, 191, 196, 197, 192, 194, 186, 191, 192, 187
6	225, 226, 221, 221, 217, 220, 221, 216, 216, 212, 219, 211, 216, 217, 212, 214, 206, 211, 212, 207
7	245, 246, 241, 241, 237, 240, 241, 236, 236, 232, 239, 231, 236, 237, 232, 234, 226, 231, 232, 227
8	265, 266, 261, 261, 257, 260, 261, 256, 256, 252, 259, 251, 256, 257, 252, 254, 246, 251, 252, 247
9	285, 286, 281, 281, 277, 280, 281, 276, 276, 272, 279, 271, 276, 277, 272, 274, 266, 271, 272, 267
10	305, 306, 301, 301, 297, 300, 301, 296, 296, 292, 299, 291, 296, 297, 292, 294, 286, 291, 292, 287
11	325, 326, 321, 321, 317, 320, 321, 316, 316, 312, 319, 311, 316, 317, 312, 314, 306, 311, 312, 307
12	345, 346, 341, 341, 337, 340, 341, 336, 336, 332, 339, 331, 336, 337, 332, 334, 326, 331, 332, 327
13	365, 366, 361, 361, 357, 360, 361, 356, 356, 352, 359, 351, 356, 357, 352, 354, 346, 351, 352, 347
14	385, 386, 381, 381, 377, 380, 381, 376, 376, 372, 379, 371, 376, 377, 372, 374, 366, 371, 372, 367
15	405, 406, 401, 401, 397, 400, 401, 396, 396, 392, 399, 391, 396, 397, 392, 394, 386, 391, 392, 387
16	425, 426, 421, 421, 417, 420, 421, 416, 416, 412, 419, 411, 416, 417, 412, 414, 406, 411, 412, 407
17	445, 446, 441, 441, 437, 440, 441, 436, 436, 432, 439, 431, 436, 437, 432, 434, 426, 431, 432, 427
18	465, 466, 461, 461, 457, 460, 461, 456, 456, 452, 459, 451, 456, 457, 452, 454, 446, 451, 452, 447
19	485, 486, 481, 481, 477, 480, 481, 476, 476, 472, 479, 471, 476, 477, 472, 474, 466, 471, 472, 467
20	505, 506, 501, 501, 497, 500, 501, 496, 496, 492, 499, 491, 496, 497, 492, 494, 486, 491, 492, 487
21	525, 526, 521, 521, 517, 520, 521, 516, 516, 512, 519, 511, 516, 517, 512, 514, 506, 511, 512, 507
22	545, 546, 541, 541, 537, 540, 541, 536, 536, 532, 539, 531, 536, 537, 532, 534, 526, 531, 532, 527
23	565, 566, 561, 561, 557, 560, 561, 556, 556, 552, 559, 551, 556, 557, 552, 554, 546, 551, 552, 547
24	585, 586, 581, 581, 577, 580, 581, 576, 576, 572, 579, 571, 576, 577, 572, 574, 566, 571, 572, 567

25	605, 606, 601, 601, 597, 600, 601, 596, 596, 592, 599, 591, 596, 597, 592, 594, 586, 591, 592, 587
26	625, 626, 621, 621, 617, 620, 621, 616, 616, 612, 619, 611, 616, 617, 612, 614, 606, 611, 612, 607
27	645, 646, 641, 641, 637, 640, 641, 636, 636, 632, 639, 631, 636, 637, 632, 634, 626, 631, 632, 627
28	665, 666, 661, 661, 657, 660, 661, 656, 656, 652, 659, 651, 656, 657, 652, 654, 646, 651, 652, 647
29	685, 686, 681, 681, 677, 680, 681, 676, 676, 672, 679, 671, 676, 677, 672, 674, 666, 671, 672, 667
30	705, 706, 701, 701, 697, 700, 701, 696, 696, 692, 699, 691, 696, 697, 692, 694, 686, 691, 692, 687

Приклад коду

Наведений зразок коду реалізовує завдання для даних, що подані в наступній таблиці:

Дані для прикладу	5, 7, 3, 4, 1, 9, 2, 8, 10, 6
Означення	input_data = [5, 7, 3, 4, 1, 9, 2, 8, 10, 6]

Лістинг

```
import System.IO

input_data = [5, 7, 3, 4, 1, 9, 2, 8, 10, 6]

quicksort :: ( Ord a ) => [ a ] -> [ a ]
quicksort [] = []
quicksort ( x : xs ) =
    let lesserPart = [ a | a <- xs , a <= x ]
        greaterPart = [ a | a <- xs , a > x ]
    in quicksort lesserPart ++ [ x ] ++ quicksort greaterPart

printElements :: [Int] -> IO()
printElements [] = return ()
printElements (x:xs) = do
    putStr (show x ++ " ")
    printElements xs

main :: IO ()
main = do
    let input = input_data
    let output = quicksort input
    putStr "input:\n"
    printElements input
    putStr "\n"
    putStr "output:\n"
    printElements output
    putStr "\n"
```