|  |
| --- |
| 0hnQGAuNPo4wojnr77c2x8cWiaT70FGHBZ2h-3__FYQ.png |
| Selection Sort |
| Data Structures Made Easy |
|  |

Dublin City university

# 1. *Selection Sort*

class selection\_Sort{

private static void swap(int [] selection\_Array, int index, int minimum){

int temporary = selection\_Array[index];

selection\_Array[index] = selection\_Array[minimum];

selection\_Array[minimum] = temporary;

}

public static void selection(int [] selection\_Array, int size){

for(int index\_1 = 0; index\_1 < size - 1; index\_1++){

int minimum\_Index = index\_1;

for(int index\_2 = index\_1 + 1; index\_2 < size; index\_2++){

if(selection\_Array[minimum\_Index] > selection\_Array[index\_2])

minimum\_Index = index\_2;

}

swap(selection\_Array, index\_1, minimum\_Index);

}

}

public static void main(String [] args){

System.out.print("Enter the number of elements: ");

int size = Console.readInt();

int [] selection\_Array = new int[size];

System.out.print('\n' + "Enter the elements: ");

for(int index\_1 = 0; index\_1 < size; index\_1++){

int element = Console.readInt();

selection\_Array[index\_1] = element;

}

selection(selection\_Array, size);

System.out.print('\n' + "The sorted list is: ");

for(int index\_2 = 0; index\_2 < size; index\_2++)

System.out.print(selection\_Array[index\_2] + " ");

}

}