ĐẠI HỌC QUỐC GIA
ĐẠI HỌC BÁCH KHOA TP HỒ CHÍ MINH

BÀI TẬP LỚN MÔN KIẾN TRÚC MÁY TÍNH

LỚP L08 --- HK 201

ĐỀ 6

SẮP XẾP CHUỖI BẰNG GIẢI THUẬT MERGE SORT

Sinh viên thực hiện	Mã số sinh viên
Nguyễn Đăng Tú	1912384
Trần Hoàng Công Toại	1912237
Trịnh Nguyên Bảo Tuấn	1912371

Thành phố Hồ Chí Minh – 2020

1. Code:

Hàm main trong chương trình:

```
233 ###############################################################################
234 main:
235 ###############################################################################
236         # Read the array
237         jal readArray
238         beq $v0, $zero, end                             # If read nothing, exit
239         # Print initial array
240         addu $a0, $v0, $zero                            # a0 = &array[0]
241         addu $a1, $v1, $zero                            # a1 = &array[last]
242         la $a2, noti1                                   # print with noti1
243         addiu $a3, $zero, '\n'                          # print '\n' at the end
244         jal printArray
245         # Sort array using mergeSort
246                                                         # Unchange a0 = first, a1 = last element of array
247         jal mergeSort
248         # Print sorted array
249                                                         # Unchange a0 = first, a1 = last element of array
250         la $a2, noti2                                   # Print with noti2
251                                                         # Unchange a3 = '\n'
252         jal printArray
253         # Unallocate memory for the array
254                                                         # Unchange a0 = first, a1 = last element of array
255         jal deleteArray
256 end:    addiu $v0, $zero, 10                            # Exit the program
257         syscall
258
```

2. Test case:

- Test case 1:

Clear

```
Initial array: -4 56 8 0 19 -31 145 7034 6 14 2020 -5 -46 10 827 -704 -1209 56091 1 -76
Divide: -4 56 8 0 19 -31 145 7034 6 14|2020 -5 -46 10 827 -704 -1209 56091 1 -76
Divide: -4 56 8 0 19|-31 145 7034 6 14
Divide: -4 56 8|0 19
Divide: -4 56|8
Divide: -4|56
Merge: -4&56
-----> -4 56
Merge: -4 56&8
-----> -4 8 56
Divide: 0|19
Merge: 0&19
-----> 0 19
Merge: -4 8 56&0 19
-----> -4 0 8 19 56
Divide: -31 145 7034|6 14
Divide: -31 145|7034
Divide: -31|145
Merge: -31&145
-----> -31 145
Merge: -31 145&7034
-----> -31 145 7034
Divide: 6|14
Merge: 6&14
-----> 6 14
Merge: -31 145 7034&6 14
-----> -31 6 14 145 7034
Merge: -4 0 8 19 56&-31 6 14 145 7034
-----> -31 -4 0 6 8 14 19 56 145 7034
```

Clear

```
Divide: 2020 -5 -46 10 827|-704 -1209 56091 1 -76
Divide: 2020 -5 -46|10 827
Divide: 2020 -5|-46
Divide: 2020|-5
Merge: 2020&-5
-----> -5 2020
Merge: -5 2020&-46
-----> -46 -5 2020
Divide: 10|827
Merge: 10&827
-----> 10 827
Merge: -46 -5 2020&10 827
-----> -46 -5 10 827 2020
Divide: -704 -1209 56091|1 -76
Divide: -704 -1209|56091
Divide: -704|-1209
Merge: -704&-1209
-----> -1209 -704
Merge: -1209 -704&56091
-----> -1209 -704 56091
Divide: 1|-76
Merge: 1&-76
-----> -76 1
Merge: -1209 -704 56091&-76 1
-----> -1209 -704 -76 1 56091
Merge: -46 -5 10 827 2020&-1209 -704 -76 1 56091
-----> -1209 -704 -76 -46 -5 1 10 827 2020 56091
Merge: -31 -4 0 6 8 14 19 56 145 7034&-1209 -704 -76 -46 -5 1 10 827 2020 56091
-----> -1209 -704 -76 -46 -31 -5 -4 0 1 6 8 10 14 19 56 145 827 2020 7034 56091
Sorted array: -1209 -704 -76 -46 -31 -5 -4 0 1 6 8 10 14 19 56 145 827 2020 7034 56091

-- program is finished running --
```

- Test case 2:

Clear

```
Initial array: -123456 12345 23 0 -234 435 564 1 -276 -457 6722 233 45 3456 12 -233 -146 -5 6 2
Divide: -123456 12345 23 0 -234 435 564 1 -276 -457|6722 233 45 3456 12 -233 -146 -5 6 2
Divide: -123456 12345 23 0 -234|435 564 1 -276 -457
Divide: -123456 12345 23|0 -234
Divide: -123456 12345|23
Divide: -123456|12345
Merge: -123456&12345
-----> -123456 12345
Merge: -123456 12345&23
-----> -123456 23 12345
Divide: 0|-234
Merge: 0&-234
-----> -234 0
Merge: -123456 23 12345&-234 0
-----> -123456 -234 0 23 12345
Divide: 435 564 1|-276 -457
Divide: 435 564|1
Divide: 435|564
Merge: 435&564
-----> 435 564
Merge: 435 564&1
-----> 1 435 564
Divide: -276|-457
Merge: -276&-457
-----> -457 -276
Merge: 1 435 564&-457 -276
-----> -457 -276 1 435 564
Merge: -123456 -234 0 23 12345&-457 -276 1 435 564
-----> -123456 -457 -276 -234 0 1 23 435 564 12345
```

Clear

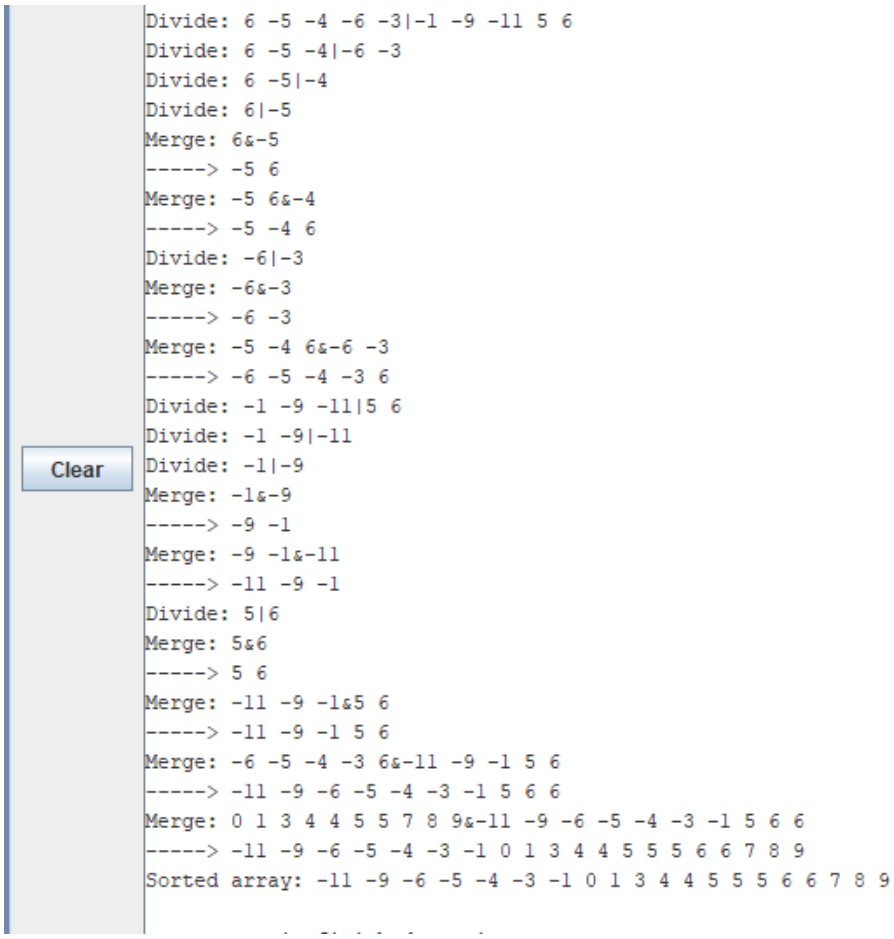
```
Divide: 6722 233 45 3456 12|-233 -146 -5 6 2
Divide: 6722 233 45|3456 12
Divide: 6722 233|45
Divide: 6722|233
Merge: 6722&233
-----> 233 6722
Merge: 233 6722&45
-----> 45 233 6722
Divide: 3456|12
Merge: 3456&12
-----> 12 3456
Merge: 45 233 6722&12 3456
-----> 12 45 233 3456 6722
Divide: -233 -146 -5|6 2
Divide: -233 -146|-5
Divide: -233|-146
Merge: -233&-146
-----> -233 -146
Merge: -233 -146&-5
-----> -233 -146 -5
Divide: 6|2
Merge: 6&2
-----> 2 6
Merge: -233 -146 -5&2 6
-----> -233 -146 -5 2 6
Merge: 12 45 233 3456 6722&-233 -146 -5 2 6
-----> -233 -146 -5 2 6 12 45 233 3456 6722
Merge: -123456 -457 -276 -234 0 1 23 435 564 12345&-233 -146 -5 2 6 12 45 233 3456 6722
-----> -123456 -457 -276 -234 -233 -146 -5 0 1 2 6 12 23 45 233 435 564 3456 6722 12345
Sorted array: -123456 -457 -276 -234 -233 -146 -5 0 1 2 6 12 23 45 233 435 564 3456 6722 12345

-- program is finished running --
```

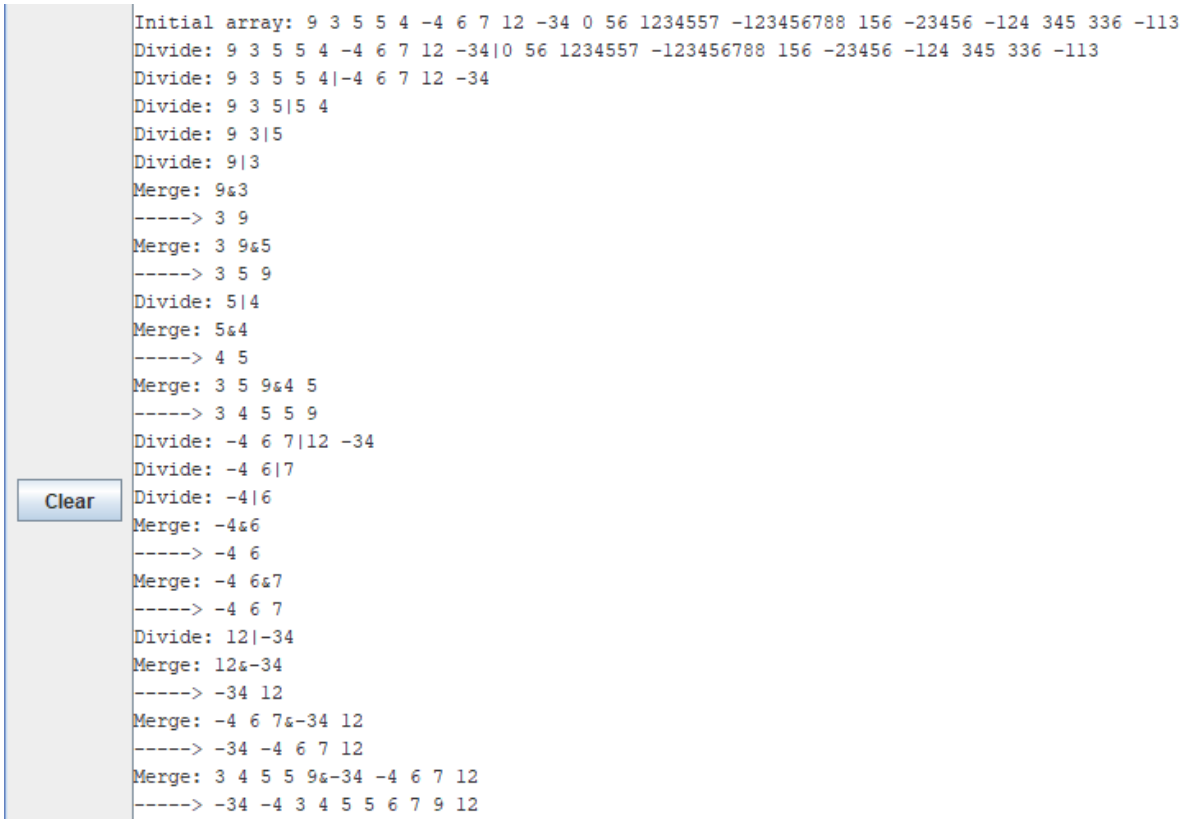
- Test case 3:

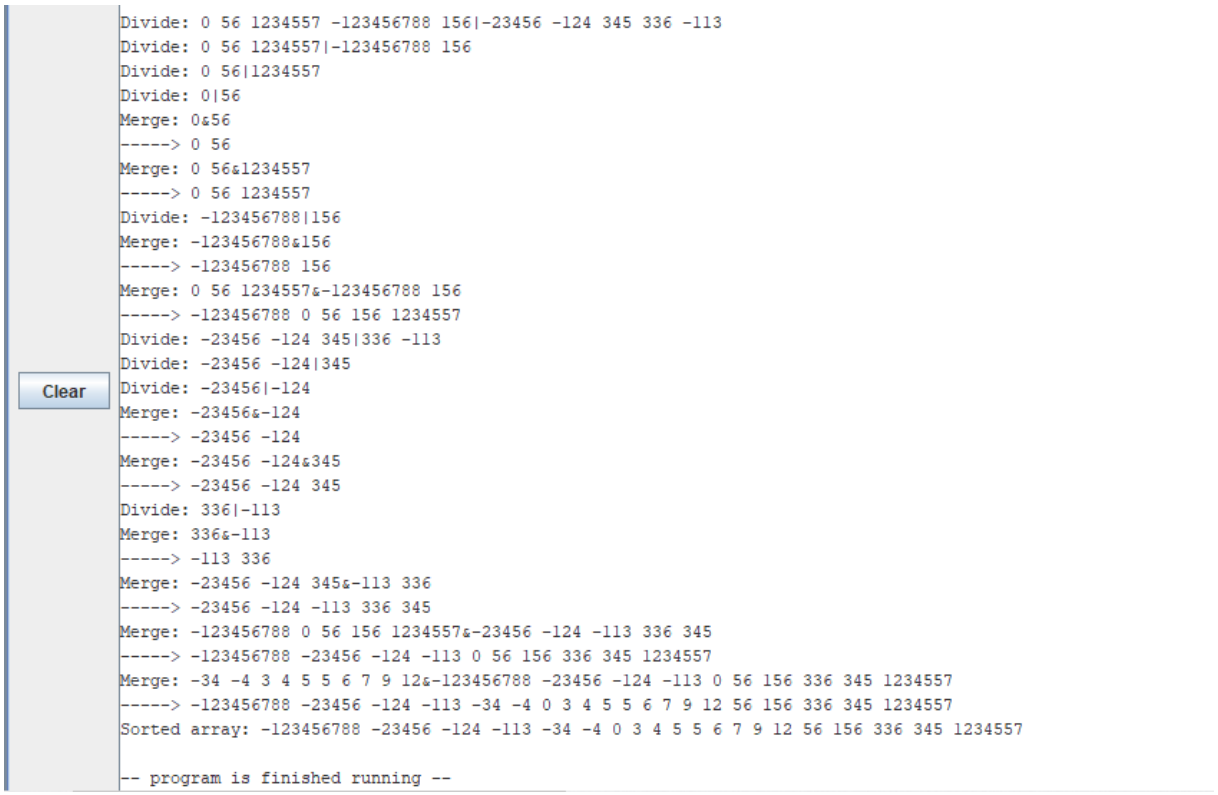
Clear

```
Initial array: 0 1 4 5 3 8 7 9 4 5 6 -5 -4 -6 -3 -1 -9 -11 5 6
Divide: 0 1 4 5 3 8 7 9 4 5|6 -5 -4 -6 -3 -1 -9 -11 5 6
Divide: 0 1 4 5 3|8 7 9 4 5
Divide: 0 1 4|5 3
Divide: 0 1|4
Divide: 0|1
Merge: 0&1
-----> 0 1
Merge: 0 1&4
-----> 0 1 4
Divide: 5|3
Merge: 5&3
-----> 3 5
Merge: 0 1 4&3 5
-----> 0 1 3 4 5
Divide: 8 7 9|4 5
Divide: 8 7|9
Divide: 8|7
Merge: 8&7
-----> 7 8
Merge: 7 8&9
-----> 7 8 9
Divide: 4|5
Merge: 4&5
-----> 4 5
Merge: 7 8 9&4 5
-----> 4 5 7 8 9
Merge: 0 1 3 4 5&4 5 7 8 9
-----> 0 1 3 4 4 5 5 7 8 9
```

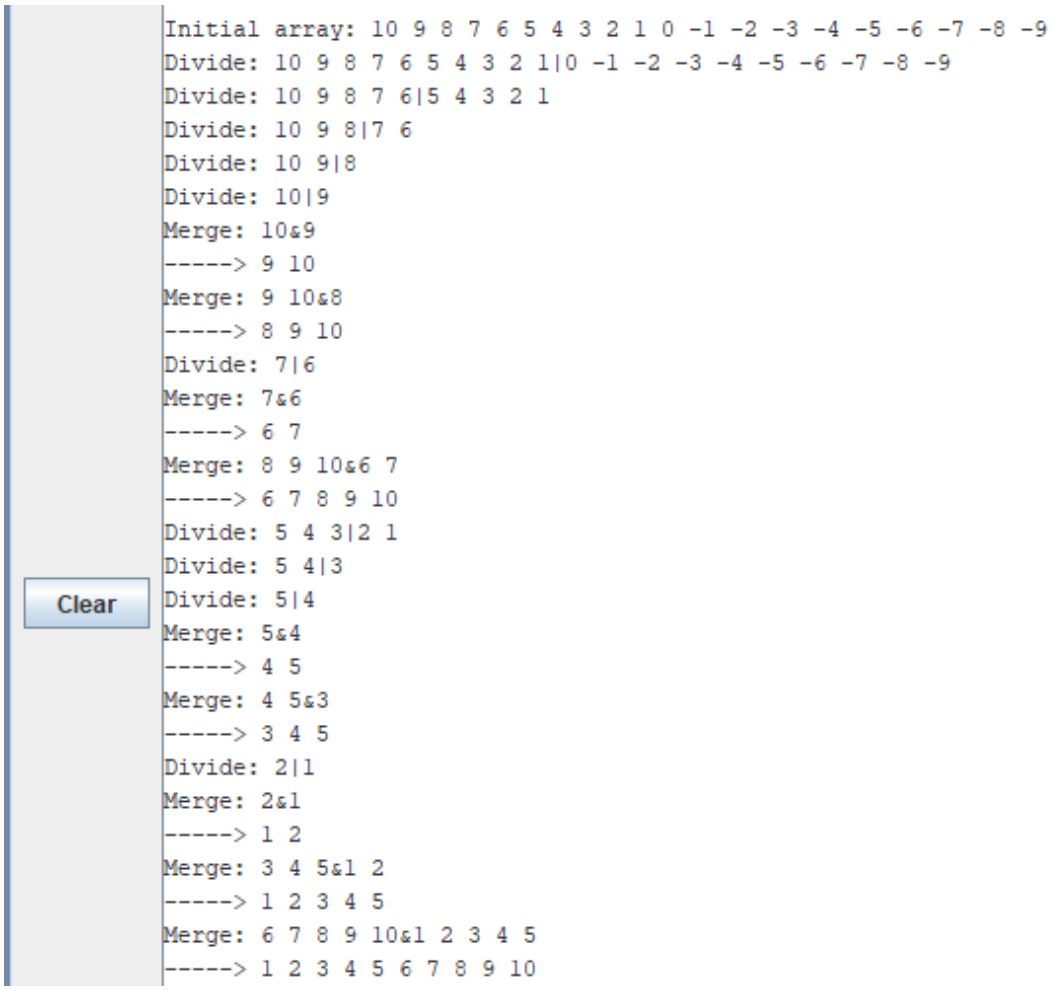

Clear
Divide: 6 -5 -4 -6 -3|-1 -9 -11 5 6
Divide: 6 -5 -4|-6 -3
Divide: 6 -5|-4
Divide: 6|-5
Merge: 6&-5
-----> -5 6
Merge: -5 6&-4
-----> -5 -4 6
Divide: -6|-3
Merge: -6&-3
-----> -6 -3
Merge: -5 -4 6&-6 -3
-----> -6 -5 -4 -3 6
Divide: -1 -9 -11|5 6
Divide: -1 -9|-11
Divide: -1|-9
Merge: -1&-9
-----> -9 -1
Merge: -9 -1&-11
-----> -11 -9 -1
Divide: 5|6
Merge: 5&6
-----> 5 6
Merge: -11 -9 -1&5 6
-----> -11 -9 -1 5 6
Merge: -6 -5 -4 -3 6&-11 -9 -1 5 6
-----> -11 -9 -6 -5 -4 -3 -1 5 6 6
Merge: 0 1 3 4 4 5 5 7 8 9&-11 -9 -6 -5 -4 -3 -1 5 6 6
-----> -11 -9 -6 -5 -4 -3 -1 0 1 3 4 4 5 5 5 6 6 7 8 9
Sorted array: -11 -9 -6 -5 -4 -3 -1 0 1 3 4 4 5 5 5 6 6 7 8 9

- Test case 4:


Clear
Initial array: 9 3 5 5 4 -4 6 7 12 -34 0 56 1234557 -123456788 156 -23456 -124 345 336 -113
Divide: 9 3 5 5 4 -4 6 7 12 -34|0 56 1234557 -123456788 156 -23456 -124 345 336 -113
Divide: 9 3 5 5 4|-4 6 7 12 -34
Divide: 9 3 5|5 4
Divide: 9 3|5
Divide: 9|3
Merge: 9&3
-----> 3 9
Merge: 3 9&5
-----> 3 5 9
Divide: 5|4
Merge: 5&4
-----> 4 5
Merge: 3 5 9&4 5
-----> 3 4 5 5 9
Divide: -4 6 7|12 -34
Divide: -4 6|7
Divide: -4|6
Merge: -4&6
-----> -4 6
Merge: -4 6&7
-----> -4 6 7
Divide: 12|-34
Merge: 12&-34
-----> -34 12
Merge: -4 6 7&-34 12
-----> -34 -4 6 7 12
Merge: 3 4 5 5 9&-34 -4 6 7 12
-----> -34 -4 3 4 5 5 6 7 9 12


Clear
Divide: 0 56 1234557 -123456788 156|-23456 -124 345 336 -113
Divide: 0 56 1234557|-123456788 156
Divide: 0 56|1234557
Divide: 0|56
Merge: 0&56
-----> 0 56
Merge: 0 56&1234557
-----> 0 56 1234557
Divide: -123456788|156
Merge: -123456788&156
-----> -123456788 156
Merge: 0 56 1234557&-123456788 156
-----> -123456788 0 56 156 1234557
Divide: -23456 -124 345|336 -113
Divide: -23456 -124|345
Divide: -23456|-124
Merge: -23456&-124
-----> -23456 -124
Merge: -23456 -124&345
-----> -23456 -124 345
Divide: 336|-113
Merge: 336&-113
-----> -113 336
Merge: -23456 -124 345&-113 336
-----> -23456 -124 -113 336 345
Merge: -123456788 0 56 156 1234557&-23456 -124 -113 336 345
-----> -123456788 -23456 -124 -113 0 56 156 336 345 1234557
Merge: -34 -4 3 4 5 5 6 7 9 12&-123456788 -23456 -124 -113 0 56 156 336 345 1234557
-----> -123456788 -23456 -124 -113 -34 -4 0 3 4 5 5 6 7 9 12 56 156 336 345 1234557
Sorted array: -123456788 -23456 -124 -113 -34 -4 0 3 4 5 5 6 7 9 12 56 156 336 345 1234557
-- program is finished running --

- Test case 5:


Clear
Initial array: 10 9 8 7 6 5 4 3 2 1 0 -1 -2 -3 -4 -5 -6 -7 -8 -9
Divide: 10 9 8 7 6 5 4 3 2 1|0 -1 -2 -3 -4 -5 -6 -7 -8 -9
Divide: 10 9 8 7 6|5 4 3 2 1
Divide: 10 9 8|7 6
Divide: 10 9|8
Divide: 10|9
Merge: 10&9
-----> 9 10
Merge: 9 10&8
-----> 8 9 10
Divide: 7|6
Merge: 7&6
-----> 6 7
Merge: 8 9 10&6 7
-----> 6 7 8 9 10
Divide: 5 4 3|2 1
Divide: 5 4|3
Divide: 5|4
Merge: 5&4
-----> 4 5
Merge: 4 5&3
-----> 3 4 5
Divide: 2|1
Merge: 2&1
-----> 1 2
Merge: 3 4 5&1 2
-----> 1 2 3 4 5
Merge: 6 7 8 9 10&1 2 3 4 5
-----> 1 2 3 4 5 6 7 8 9 10

Clear

```
Divide: 0 -1 -2 -3 -4|-5 -6 -7 -8 -9
Divide: 0 -1 -2|-3 -4
Divide: 0 -1|-2
Divide: 0|-1
Merge: 0&-1
-----> -1 0
Merge: -1 0&-2
-----> -2 -1 0
Divide: -3|-4
Merge: -3&-4
-----> -4 -3
Merge: -2 -1 0&-4 -3
-----> -4 -3 -2 -1 0
Divide: -5 -6 -7|-8 -9
Divide: -5 -6|-7
Divide: -5|-6
Merge: -5&-6
-----> -6 -5
Merge: -6 -5&-7
-----> -7 -6 -5
Divide: -8|-9
Merge: -8&-9
-----> -9 -8
Merge: -7 -6 -5&-9 -8
-----> -9 -8 -7 -6 -5
Merge: -4 -3 -2 -1 0&-9 -8 -7 -6 -5
-----> -9 -8 -7 -6 -5 -4 -3 -2 -1 0
Merge: 1 2 3 4 5 6 7 8 9 10&-9 -8 -7 -6 -5 -4 -3 -2 -1 0
-----> -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 10
Sorted array: -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 10

-- program is finished running --
```

- Test case 6:

Clear

```
Initial array: 1 3 5 7 9 2 4 6 8 10 11 13 15 17 19 12 14 16 18 20
Divide: 1 3 5 7 9 2 4 6 8 10|11 13 15 17 19 12 14 16 18 20
Divide: 1 3 5 7 9|2 4 6 8 10
Divide: 1 3 5|7 9
Divide: 1 3|5
Divide: 1|3
Merge: 1&3
-----> 1 3
Merge: 1 3&5
-----> 1 3 5
Divide: 7|9
Merge: 7&9
-----> 7 9
Merge: 1 3 5&7 9
-----> 1 3 5 7 9
Divide: 2 4 6|8 10
Divide: 2 4|6
Divide: 2|4
Merge: 2&4
-----> 2 4
Merge: 2 4&6
-----> 2 4 6
Divide: 8|10
Merge: 8&10
-----> 8 10
Merge: 2 4 6&8 10
-----> 2 4 6 8 10
Merge: 1 3 5 7 9&2 4 6 8 10
-----> 1 2 3 4 5 6 7 8 9 10
```

Clear

```
Divide: 11 13 15 17 19|12 14 16 18 20
Divide: 11 13 15|17 19
Divide: 11 13|15
Divide: 11|13
Merge: 11&13
-----> 11 13
Merge: 11 13&15
-----> 11 13 15
Divide: 17|19
Merge: 17&19
-----> 17 19
Merge: 11 13 15&17 19
-----> 11 13 15 17 19
Divide: 12 14 16|18 20
Divide: 12 14|16
Divide: 12|14
Merge: 12&14
-----> 12 14
Merge: 12 14&16
-----> 12 14 16
Divide: 18|20
Merge: 18&20
-----> 18 20
Merge: 12 14 16&18 20
-----> 12 14 16 18 20
Merge: 11 13 15 17 19&12 14 16 18 20
-----> 11 12 13 14 15 16 17 18 19 20
Merge: 1 2 3 4 5 6 7 8 9 10&11 12 13 14 15 16 17 18 19 20
-----> 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Sorted array: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

-- program is finished running --
```

- Test case 7:

Clear

```
Initial array: 2 2 2 2 1 1 1 1 3 3 3 -1 -1 0 0 0 5 5 6 6
Divide: 2 2 2 2 1 1 1 1 3 3|3 -1 -1 0 0 0 5 5 6 6
Divide: 2 2 2 2 1|1 1 1 3 3
Divide: 2 2 2|2 1
Divide: 2 2|2
Divide: 2|2
Merge: 2&2
-----> 2 2
Merge: 2 2&2
-----> 2 2 2
Divide: 2|1
Merge: 2&1
-----> 1 2
Merge: 2 2 2&1 2
-----> 1 2 2 2 2
Divide: 1 1 1|3 3
Divide: 1 1|1
Divide: 1|1
Merge: 1&1
-----> 1 1
Merge: 1 1&1
-----> 1 1 1
Divide: 3|3
Merge: 3&3
-----> 3 3
Merge: 1 1 1&3 3
-----> 1 1 1 3 3
Merge: 1 2 2 2 2&1 1 1 3 3
-----> 1 1 1 1 2 2 2 2 3 3
```

Clear

```
Divide: 3 -1 -1 0 0|0 5 5 6 6
Divide: 3 -1 -1|0 0
Divide: 3 -1|-1
Divide: 3|-1
Merge: 3&-1
-----> -1 3
Merge: -1 3&-1
-----> -1 -1 3
Divide: 0|0
Merge: 0&0
-----> 0 0
Merge: -1 -1 3&0 0
-----> -1 -1 0 0 3
Divide: 0 5 5|6 6
Divide: 0 5|5
Divide: 0|5
Merge: 0&5
-----> 0 5
Merge: 0 5&5
-----> 0 5 5
Divide: 6|6
Merge: 6&6
-----> 6 6
Merge: 0 5 5&6 6
-----> 0 5 5 6 6
Merge: -1 -1 0 0 3&0 5 5 6 6
-----> -1 -1 0 0 0 3 5 5 6 6
Merge: 1 1 1 1 2 2 2 2 3 3&-1 -1 0 0 0 3 5 5 6 6
-----> -1 -1 0 0 0 1 1 1 1 2 2 2 2 3 3 3 5 5 6 6
Sorted array: -1 -1 0 0 0 1 1 1 1 2 2 2 2 3 3 3 5 5 6 6

-- program is finished running --
```

- Test case 8:

Clear

```
Initial array: -3 -5 -7 1 2 4 -5 -8 2 2 3 -9 -5 -7 12 20 -8 35 13 24
Divide: -3 -5 -7 1 2 4 -5 -8 2 2|3 -9 -5 -7 12 20 -8 35 13 24
Divide: -3 -5 -7 1 2|4 -5 -8 2 2
Divide: -3 -5 -7|1 2
Divide: -3 -5|-7
Divide: -3|-5
Merge: -3&-5
-----> -5 -3
Merge: -5 -3&-7
-----> -7 -5 -3
Divide: 1|2
Merge: 1&2
-----> 1 2
Merge: -7 -5 -3&1 2
-----> -7 -5 -3 1 2
Divide: 4 -5 -8|2 2
Divide: 4 -5|-8
Divide: 4|-5
Merge: 4&-5
-----> -5 4
Merge: -5 4&-8
-----> -8 -5 4
Divide: 2|2
Merge: 2&2
-----> 2 2
Merge: -8 -5 4&2 2
-----> -8 -5 2 2 4
Merge: -7 -5 -3 1 2&-8 -5 2 2 4
-----> -8 -7 -5 -5 -3 1 2 2 2 4
```

Clear

```
Divide: 3 -9 -5 -7 12|20 -8 35 13 24
Divide: 3 -9 -5|-7 12
Divide: 3 -9|-5
Divide: 3|-9
Merge: 3&-9
-----> -9 3
Merge: -9 3&-5
-----> -9 -5 3
Divide: -7|12
Merge: -7&12
-----> -7 12
Merge: -9 -5 3&-7 12
-----> -9 -7 -5 3 12
Divide: 20 -8 35|13 24
Divide: 20 -8|35
Divide: 20|-8
Merge: 20&-8
-----> -8 20
Merge: -8 20&35
-----> -8 20 35
Divide: 13|24
Merge: 13&24
-----> 13 24
Merge: -8 20 35&13 24
-----> -8 13 20 24 35
Merge: -9 -7 -5 3 12&-8 13 20 24 35
-----> -9 -8 -7 -5 3 12 13 20 24 35
Merge: -8 -7 -5 -5 -3 1 2 2 2 4&-9 -8 -7 -5 3 12 13 20 24 35
-----> -9 -8 -8 -7 -7 -5 -5 -5 -3 1 2 2 2 3 4 12 13 20 24 35
Sorted array: -9 -8 -8 -7 -7 -5 -5 -5 -3 1 2 2 2 3 4 12 13 20 24 35

-- program is finished running --
```

- Test case 9:

Clear

```
Initial array: 0 7 1 2 2 0 0 1 3 0 1 2 2 0 0 1 2 3 4 5
Divide: 0 7 1 2 2 0 0 1 3 0|1 2 2 0 0 1 2 3 4 5
Divide: 0 7 1 2 2|0 0 1 3 0
Divide: 0 7 1|2 2
Divide: 0 7|1
Divide: 0|7
Merge: 0&7
-----> 0 7
Merge: 0 7&1
-----> 0 1 7
Divide: 2|2
Merge: 2&2
-----> 2 2
Merge: 0 1 7&2 2
-----> 0 1 2 2 7
Divide: 0 0 1|3 0
Divide: 0 0|1
Divide: 0|0
Merge: 0&0
-----> 0 0
Merge: 0 0&1
-----> 0 0 1
Divide: 3|0
Merge: 3&0
-----> 0 3
Merge: 0 0 1&0 3
-----> 0 0 0 1 3
Merge: 0 1 2 2 7&0 0 0 1 3
-----> 0 0 0 0 1 1 2 2 3 7
```

Clear

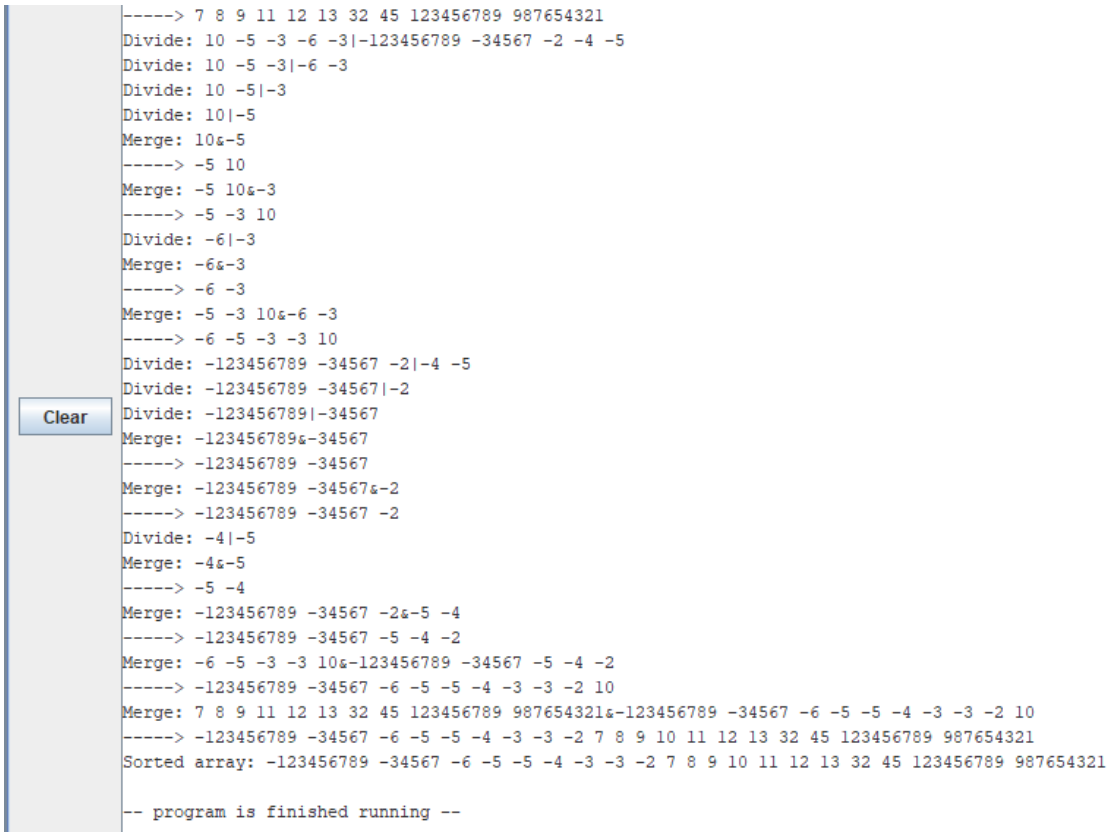
```
Divide: 1 2 2 0 0|1 2 3 4 5
Divide: 1 2 2|0 0
Divide: 1 2|2
Divide: 1|2
Merge: 1&2
-----> 1 2
Merge: 1 2&2
-----> 1 2 2
Divide: 0|0
Merge: 0&0
-----> 0 0
Merge: 1 2 2&0 0
-----> 0 0 1 2 2
Divide: 1 2 3|4 5
Divide: 1 2|3
Divide: 1|2
Merge: 1&2
-----> 1 2
Merge: 1 2&3
-----> 1 2 3
Divide: 4|5
Merge: 4&5
-----> 4 5
Merge: 1 2 3&4 5
-----> 1 2 3 4 5
Merge: 0 0 1 2 2&1 2 3 4 5
-----> 0 0 1 1 2 2 2 3 4 5
Merge: 0 0 0 0 1 1 2 2 3 7&0 0 1 1 2 2 2 3 4 5
-----> 0 0 0 0 0 0 0 1 1 1 1 2 2 2 2 2 3 3 4 5 7
Sorted array: 0 0 0 0 0 0 0 1 1 1 1 2 2 2 2 2 3 3 4 5 7

-- program is finished running --
```

- Test case 10:

Clear

```
Initial array: 123456789 9 8 7 987654321 12 13 45 32 11 10 -5 -3 -6 -3 -123456789 -34567 -2 -4 -5
Divide: 123456789 9 8 7 987654321 12 13 45 32 11|10 -5 -3 -6 -3 -123456789 -34567 -2 -4 -5
Divide: 123456789 9 8 7 987654321|12 13 45 32 11
Divide: 123456789 9 8|7 987654321
Divide: 123456789 9|8
Divide: 123456789|9
Merge: 123456789&9
-----> 9 123456789
Merge: 9 123456789&8
-----> 8 9 123456789
Divide: 7|987654321
Merge: 7&987654321
-----> 7 987654321
Merge: 8 9 123456789&7 987654321
-----> 7 8 9 123456789 987654321
Divide: 12 13 45|32 11
Divide: 12 13|45
Divide: 12|13
Merge: 12&13
-----> 12 13
Merge: 12 13&45
-----> 12 13 45
Divide: 32|11
Merge: 32&11
-----> 11 32
Merge: 12 13 45&11 32
-----> 11 12 13 32 45
Merge: 7 8 9 123456789 987654321&11 12 13 32 45
-----> 7 8 9 11 12 13 32 45 123456789 987654321
```



3. Thống kê số lệnh, loại lệnh:

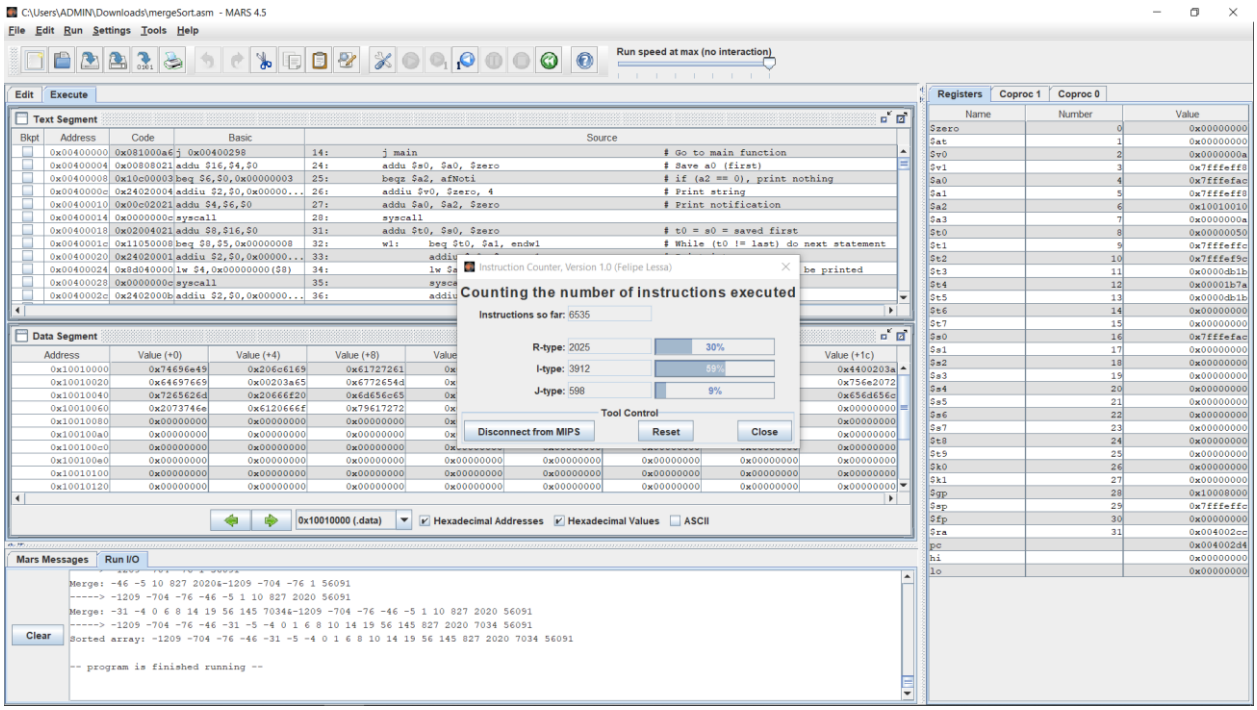
Sử dụng test case sau:

Initial array: -4 56 8 0 19 -31 145 7034 6 14 2020 -5 -46 10 827 -704 -1209 56091 1 -76

Sorted array: -1209 -704 -76 -46 -31 -5 -4 0 1 6 8 10 14 19 56 145 827 2020 7034 56091

- Tính số lệnh và phân loại theo định dạng R, I, J, sử dụng

$Tools \rightarrow Instruction\ Counter$:



Tổng số lệnh: 6535

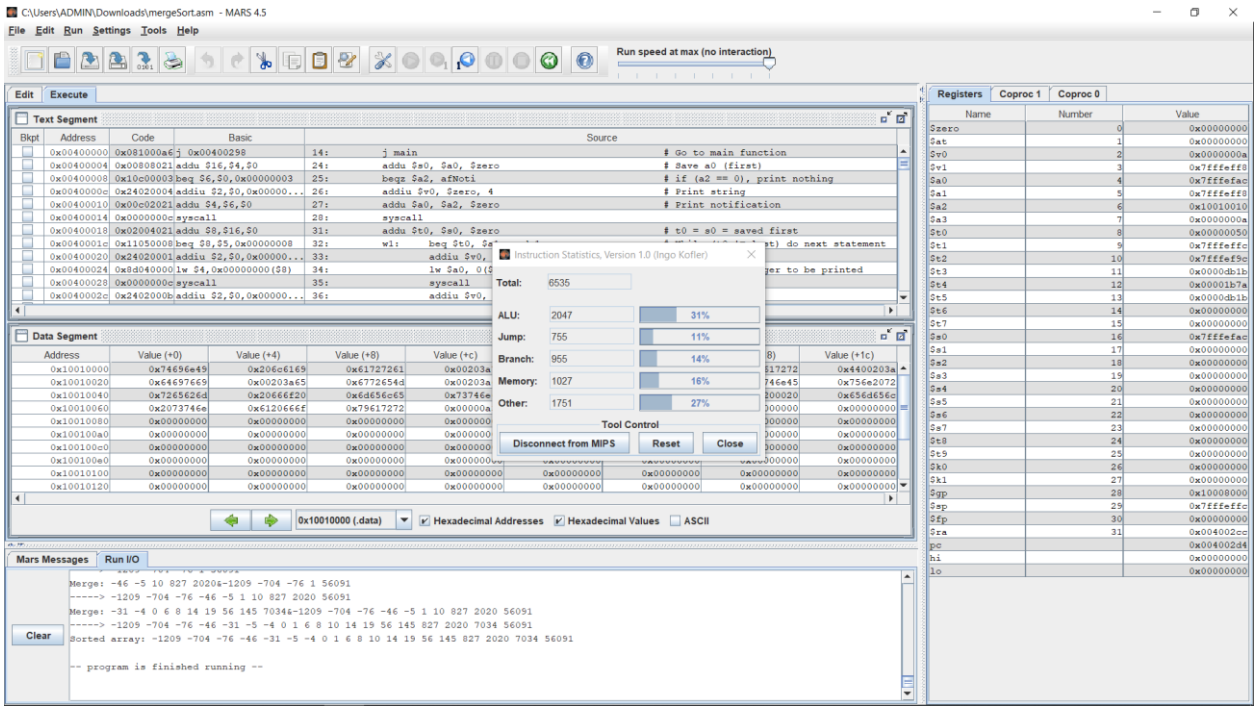
Số lệnh dạng R: 2025

Số lệnh dạng I: 3912

Số lệnh dạng J: 598

- Tính số lệnh và phân loại theo chức năng, sử dụng

Tools → Instruction Statistics:



Tổng số lệnh: 6535

Số lệnh ALU: 2047

Số lệnh Jump: 755

Số lệnh Branch: 955

Số lệnh Memory: 1027

Số lệnh khác: 1751

4. Tính thời gian chạy của chương trình:

Thời gian chạy chương trình được xác định thông qua thời gian thực thi của CPU (CPU time).

CPU time được tính theo công thức:

$$\begin{aligned}\text{CPU time} &= \text{CPU clock cycles x Clock cycle time} \\ &= \text{Instruction count x CPI x Clock cycle time} \\ &= \frac{\text{Instruction count x CPI}}{\text{Clock rate}}\end{aligned}$$

Trong đó:

CPU time : thời gian thực thi của CPU

CPU clock cycles : số chu kỳ của chương trình

Clock cycle time : thời gian một chu kỳ xung nhịp

Instruction count : số lệnh thực thi

CPI : số chu kỳ của mỗi lệnh

Clock rate : tần số xung nhịp

Dựa vào công thức trên với:

Instruction count = 6535

CPI = 1

Clock rate = 2GHz

Ta có:

$$\text{CPU time} = \frac{(6535 \text{ x } 1)}{(2 \text{ x } 10^9)} = 3267{,}5 \text{ x } 10^{-9}(\text{s}) = 3267{,}5(\text{ns})$$