



### تمرین Iterative Merge Sort:

هدف از این تمرین پیاده سازی الگوریتم Iterative Merge Sort به زبان جاوا است. در این تمرین از دانشجو خواسته شده است که متدهای زیر را در فایل MergeSort.java پیاده سازی کند:

1-متد sort:

```
public void sort(T[] data, int n){
    //Write your code here
}
```

این متد، بخش اصلی کد شما است و در آن کد اصلی مرتب سازی به جز بخش merge آن پیاده سازی می شود. این متد یک آرایه data از نوع جنریک T دریافت میکند که این نوع داده، اینترفیس Comparable را پیاده سازی کرده است. همچنین این متد اندازه آرایه n - n را نیز دریافت می کند.

2- متد merge:

```
public void merge (T[] arr, T[] temp, int p, int q, int r){
    //Write your code here
}
```

این متد با دریافت آرایه arr، آرایه temp و نقاط مشخص شده p، q و r دو بخش آرایه را که با توجه به نقاط مشخص شده اند را باهم ادغام می کند.

\* تابع ادغام به طور کامل در درس شرح داده شده است.

در این تمرین نحوه ارزیابی به شرح زیر است:

1. TestMerge: این تست، متد merge شما را مورد سنجش قرار می دهد. انتظار می رود تا کد نوشته شده شما بتواند در آرایه ای به اندازه بزرگ نیز به درستی کار کند. این تست ۲۵ درصد از نمره تمرین را به خود اختصاص می دهد.
2. TestRandomNumbers: در این تست، درستی کد شما با آرایه هایی با اندازه های بالا و با مقادیر رندوم مورد سنجش قرار می گیرد. این تست ۳۵ درصد از نمره تمرین را به خود اختصاص می دهد.
3. TestOrder: در این تست، کد شما با آرایه ای با اندازه ی بسیار بزرگ، از نظر زمانی و اوردری مورد سنجش قرار می گیرد. در صورت بهینه بودن کد شما و داشتن اوردر  $O(n \log n)$ ، شما نمره ۱۰۰ را از این تست کسب خواهید کرد و در غیر این صورت نمره ای به شما از این تست تعلق نخواهد گرفت. این تست ۴۰ درصد از نمره این تمرین را به خود اختصاص می دهد.

برای انجام این تمرین کارهای زیر را انجام دهید:

- 1- ابتدا در این پوشه فایل info.txt را با مشخصات خود پر کنید.
- 2- سپس فایل پروژه را در IntelliJ باز کنید.
- 3- تمرین را انجام دهید و بخش‌های ناقص کد را تکمیل کنید و از درستی پیاده‌سازی خود مطمئن شوید.
- 4- محتویات کل پوشه را به صورت یک فایل zip در آورید.
- 5- مطمئن شوید که وقتی فایل zip را باز می‌کنید پوشه‌های img, src و test و همچنین فایل info.txt را می‌بینید.
- 6- نام این فایل zip را به «شماره تمرین-شماره دانشجویی» تغییر دهید (مثال: IterativeMergeSort.zip-123456789)
- 7- دقت کنید که پسوند فایل شما حتماً zip باشد.
- 8- ابتدا این فایل را به سیستم «سپهر» ایمیل کنید تا از نحوه عمل‌کرد برنامه خود بر روی تست‌های تکمیلی آگاه شوید.
- 9- اشکالاتی را که سیستم «سپهر» مشخص کرده است برطرف نمایید و مجدداً تمرین را به «سپهر» تحویل دهید.
- 10- مرحله قبل را آن‌قدر ادامه دهید که از صحت عمل‌کرد برنامه خود اطمینان حاصل نمایید.
- 11- پس از اطمینان از دریافت نمره کامل، همان فایل ارسالی را از طریق سیستم vu تحویل دهید.

با آرزوی موفقیت