به نام خدا

|  |  |  |  |
| --- | --- | --- | --- |
| مدرس: مظفر بگ محمدي | دانشگاه ايلام | ترم اول سال تحصيلي 01-00 | سري چهارم تمرينات ساختمان داده |

1. لیست دوپیوندی: با استفاده از کلاس LinkedList موجود در جاوا یک پشته و یک صف درست کنید و آنها را به صورت مناسب آزمایش کنید.
2. لیست دوپیوندی: یک لیست دوپیوندی ایجاد کنید که در هر کدام از نودهای آن یک عدد طبیعی قرار گیرد. سپس متدهای زیر را پیاده و آزمایش کنید:
   1. متد reverse که ترتیب عناصر لیست را معکوس می کند. مثلاً اگر لیست برابر 1🡪5🡪3🡪10 باشد به 10🡪3🡪5🡪1 تبدیل خواهد شد.
   2. متد display برای نمایش عناصر لیست از ابتدا به انتها.
   3. متد removegt(int a) برای حذف تمام عناصر لیست که از a بزرگتر هستند. دقت کنید که ترتیب بقیه ی عناصر نباید به هم بخورد.
   4. متد count برای شمارش تعداد اعضای لیست.
3. لیست پیوندی: یک لیست تک پیوندی ایجاد کنید که دارای متغیر head باشد و در هر کدام از نودهای آن یک شی قرار گیرد. سپس متدهای زیر را پیاده و آزمایش کنید:
   1. متدهای isEmpty( ) و size( ) که به ترتیب خالی بودن لیست را چک و اندازه ی لیست را محاسبه می کنند/
   2. متد get( int index) که شی موجود در محل index را بر می گرداند.
   3. متدهای add(int index, Object item) و remove( int index) و removeAll( )برای اضافه کردن و یا حذف اشیاء.
   4. متد swap(int i, int j ) برای جابجایی اشیاء محل i و .j
4. بازگشت: یک متد بازگشتی بنویسید که ترتیب عناصر یک لیست تک پیوندی را معکوس کند.
5. لیست دوپیوندی: یک لیست دوپیوندی ایجاد کنید و در اثر فراخوانی toString تمام اعداد موجود در لیست را که مجموع دوبه دوی آنها W می شود نمایش دهد!
6. لیست پیوندی: یک لیست تک پیوندی ایجاد کنید و بعد از هر a نود ، b نود را حذف کنید
7. لیست پیوندی: برنامه‌ای بنویسید که عنصر وسط یک لیست پیوندی را بیاید پیچیدگی زمانی مساله شما باید در زمان o(n) یا بهتر باشد!
8. لیست پیوندی: در یک لیست پیوندی pاُمین آخرین عنصر لیست را با یک بار طی کردن طول لیست پیدا کنید!
9. لیست پیوندی: یک لیست تک پیوندی ایجاد کنید و عدد y را از کاربر دریافت کنید ، لیست را طوری تغییر دهید که تمام عناصر کوچکتر از y قبل از عناصر بزرگتر یا مساوی y قرار گیرند.
10. پیچیدگی: پیچیدگی تمام متدهایی که نوشته اید را محاسبه کنید.