1. **Dizi formatında** verilen **N elamandan** oluşan son seviyesinde yer alan yaprak düğümlerin **tamamıyla dolu** olduğu bilinen **Max-Heap** özelliği taşıyan bir **ağaçta** yer alan **en küçük** ve **en büyük** değeri bulan algoritmanın **akış diyagramını çizip zaman karmaşıklığını** hesaplayınız.
2. **Bir** matris üzerinde tutulan bilgilerin yarısından fazlası "0" ise bu matris **"sparse matris"** olarak adlandırılır. Bu tür matrislerin bilgileri, kullanılan bellek miktarını azaltmak için **linkli liste yapısı** kullanılarak saklanır. **0’dan farklı olan matris hücrelerinin satır, sütun** ve **değer bilgisi**, linkli liste elemanlarında tutulur. **Linkli liste yapısında verilen NxM'lik bir sparse matris bilgilerini kullanarak** hangi **sütunlarının** tamamen "0"lardan oluştuğunu bulan programı C dilinde kodlayınız. (**Önemli:** Matris bilgisinden linkli liste dönüşümü kodunu yazmayınız. Bilgiler linkli liste formatında okunmalıdır.)

**Örnek:** Aşağıda açık hali verilen matrisin linkli listeye dönüştürülmüş hali girdi olarak verildiğinde programınızın **0,3,4** çıktısını üretmesi beklenmektedir.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | 3 | 0 | **0** | **0** | 0 | 1 |
| **0** | 0 | 2 | **0** | **0** | 0 | 4 |
| **0** | 1 | 0 | **0** | **0** | 0 | 2 |
| **0** | 0 | 0 | **0** | **0** | 0 | 0 |
| **0** | 0 | 0 | **0** | **0** | 0 | 0 |