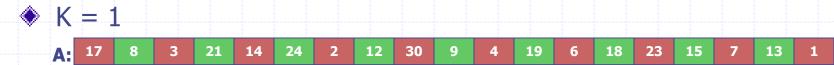
- Sortarea prin interclasare cu 3 benzi este un algoritm pentru sortarea fisierelor secventiale
- Foloseste intens algoritmul de interclasare simpla a doua secvente ordonate
- Algoritmul se poate folosi si pentru sortarea tablourilor, dar cu putin succes, deoarece este mult mai putin performant decat algoritmii existenti pentru sortarea de tablouri
- Dupa cum ii spune si numele, pentru sortare vor fi folosite in total 3 fisiere (cel de sortat plus inca 2 fisiere suplimentare pentru manevre)

Presupunem ca dorim sa sortam urmatoarea secventa:

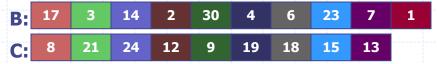
| Δ. | 17 | 8 | 3 | 21 | 14 | 24 | 2 | 12 | 30 | 9 | 4 | 19 | 6 | 18 | 23 | 15 | 7 | 13 | 1 |
|----|----|---|---|----|----|----|---|----|----|---|---|----|---|----|----|----|---|----|-----|
| | 1 | 1 | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | | 1 | 1 | 1 | 1 | 1 3 |

- Algoritmul va face mai multe treceri prin secventa (sau prin fisier, daca este vorba de un fisier)
- La fiecare trecere, fisierul de sortat este impartit in 2 fisiere mai mici, dupa care acestea vor fuziona prin interclasare pentru a forma din nou fisierul initial
- Asa se face ca in total este nevoie de 3 fisiere (benzi)
- Dupa ultima trecere, fisierul initial va fi sortat

- Algoritmul dispune de o variabila K, al carei rol este de a specifica numarul de elemente care se proceseaza deodata
- ◆ Initial, K = 1
- La fiecare trecere se executa urmatoarele activitati:
 - se copiaza cate K elemente din fisierul initial, alternativ, in cele 2 fisiere intermediare, pana la epuizarea fisierului initial
 - se reface fisierul initial din fisierele intermediare, interclasand secvente de lungime K luate din fisierele intermediare
 - K se dubleaza, pentru a procesa blocuri de lungime dubla la urmatorul pas
- Algoritmul se opreste in momentul in care K, prin dublari succesive, depaseste numarul de elemente ale secventei de sortat



Se muta alternativ cate K elemente din banda A in benzile auxiliare B si C, pana la epuizarea benzii A



- Se observa ca benzile B si C contin secvente ordonate de cate K elemente (K fiind 1 in acest moment)
- Pentru refacerea benzii A, se vor interclasa secventele corespunzatoare de lungime K din benzile B si C, adica:
 - (17) cu (8), (3) cu (21), (14) cu (24), etc.
 - se obtin secventele (8, 17), (3, 21), (14, 24), etc., care se copiaza in banda

4

K se dubleaza, devenind 2

Calin Jebelean

- Se muta alternativ cate K elemente din banda A in benzile auxiliare B si C, pana la epuizarea benzii A
 - B: 8 17 14 24 9 30 6 18 7 13 C: 3 21 2 12 4 19 15 23 1
- Se observa ca benzile B si C contin secvente ordonate de cate K elemente (K fiind 2 in acest moment)
- Pentru refacerea benzii A, se vor interclasa secventele corespunzatoare de lungime K din benzile B si C, adica:
 - (8, 17) cu (3, 21), (14, 24) cu (2, 12), (9, 30) cu (4, 19), etc.
 - se obtin secventele (3, 8, 17, 21), (2, 12, 14, 24), etc., care se copiaza in banda A
- K se dubleaza, devenind 4

Calin Jebelean

- Se muta alternativ cate K elemente din banda A in benzile auxiliare B si C, pana la epuizarea benzii A
 - B: 3 8 17 21 4 9 19 30 1 7 13 C: 2 12 14 24 6 15 18 23
- Se observa ca benzile B si C contin secvente ordonate de cate K elemente (K fiind 4 in acest moment)
- Pentru refacerea benzii A, se vor interclasa secventele corespunzatoare de lungime K din benzile B si C, adica:
 - (3, 8, 17, 21) cu (2, 12, 14, 24), (4, 9, 19, 39) cu (6, 15, 18, 23), etc.
 - se obtin secventele (2, 3, 8, 12, 14, 17, 21, 24), (4, 6, 9, 15, 18, 19, 23, 39), etc., care se copiaza in banda A
- K se dubleaza, devenind 8

- ★ K = 8
 A: 2 3 8 12 14 17 21 24 4 6 9 15 18 19 23 30 1 7 13
- Se muta alternativ cate K elemente din banda A in benzile auxiliare B si C, pana la epuizarea benzii A

```
B: 2 3 8 12 14 17 21 24 1 7 13
C: 4 6 9 15 18 19 23 30
```

- Se observa ca benzile B si C contin secvente ordonate de cate K elemente (K fiind 8 in acest moment)
- Pentru refacerea benzii A, se vor interclasa secventele corespunzatoare de lungime K din benzile B si C, adica cele colorate cu aceeasi culoare
- K se dubleaza, devenind 16

- K = 16
 A: 2 3 4 6 8 9 12 14 15 17 18 19 21 23 24 30 1 7 13
- Se muta alternativ cate K elemente din banda A in benzile auxiliare B si C, pana la epuizarea benzii A

```
B: 2 3 4 6 8 9 12 14 15 17 18 19 21 23 24 30 C: 1 7 13
```

- Se observa ca benzile B si C contin secvente ordonate de cate K elemente (K fiind 16 in acest moment)
- Pentru refacerea benzii A, se vor interclasa secventele corespunzatoare de lungime K din benzile B si C, adica cele colorate cu aceeasi culoare
- K se dubleaza, devenind 32

- K a depasit numarul de elemente ale secventei de sortat – in acest moment nu ar mai avea rost o impartire a secventei in 2 benzi
- Acest lucru este un indicator ca sortarea s-a incheiat
- Pentru depistarea mai rapida a acestei situatii, se poate contoriza cate perechi de subsecvente au fost interclasate pentru refacerea benzii A – daca acest numar este 1, algoritmul se incheie
- Secventa sortata este:

A: 1 2 3 4 6 7 8 9 12 13 14 15 17 18 19 21 23 24 30

- Datorita dublarii lui K la fiecare pas rezulta ca numarul de treceri al algoritmului este proportional cu log₂N, unde N reprezinta numarul de elemente ale secventei
- Algoritmul este oarecum asimetric, deoarece fiecare trecere poate fi defalcata in doua faze (injumatatire si interclasare) si destinatiile celor doua faze sunt benzile B si C (in cazul injumatatirii) respectiv banda A (in cazul interclasarii)
- Insusi numarul de benzi este un martor al asimetriei algoritmului
- Pe de alta parte, algoritmul este optim din punctul de vedere al numarului de benzi folosite
- Un algoritm simetric, este cel de sortare prin interclasare cu 4 benzi – acesta foloseste insa un numar mai mare de benzi