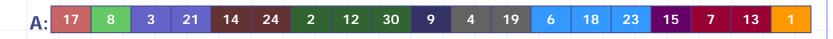
- Sortarea prin interclasare naturala este un algoritm pentru sortarea fisierelor secventiale
- Foloseste intens algoritmul de interclasare simpla a doua secvente ordonate
- Algoritmul se poate folosi si pentru sortarea tablourilor, dar cu putin succes, deoarece este mult mai putin performant decat algoritmii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

Presupunem ca dorim sa sortam urmatoarea secventa:

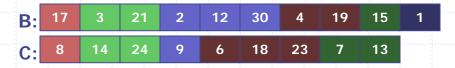
Δ.	17	8	3	21	14	24	2	12	30	9	4	19	6	18	23	15	7	13	1
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	i i	1	1	1

- 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

- La fiecare trecere se executa urmatoarele activitati:
 - se determina monotoniile din fisierul initial (o monotonie este un subsir de lungime maxima, ordonat) – evident, determinarea monotoniilor se face pe masura parcurgerii fisierului, si nu intr-un pas separat
 - se copiaza monotoniile din fisierul initial, alternativ, in cele 2 fisiere intermediare, pana la epuizarea fisierului initial
 - se reface fisierul initial din fisierele intermediare, interclasand cate o monotonie din primul fisier intermediar cu cate o monotonie din al doilea fisier intermediar
- Algoritmul se opreste in momentul in care numarul de monotonii din fisierul initial ajunge egal cu 1 – asta inseamna ca intreg fisierul este o monotonie, adica este ordonat



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



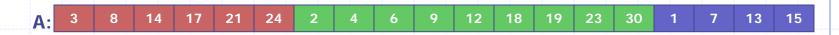
- Pentru refacerea benzii A, se vor interclasa monotoniile din benzile B si C, doua cate doua, adica:
 - (17) cu (8), (3, 21) cu (14, 24), (2, 12, 30) cu (9), etc.
 - se obtin secventele (8, 17), (3, 14, 21, 24), (2, 9, 12, 30), etc.,
 care se copiaza in banda A



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

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

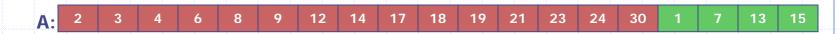
- Pentru refacerea benzii A, se vor interclasa monotoniile din benzile B si C, doua cate doua, adica:
 - (8, 17) cu (3, 14, 21, 24), (2, 9, 12, 30) cu (4, 6, 18, 19, 23), etc.
 - se obtin secventele (3, 8, 14, 17, 21, 24), (2, 4, 6, 9, 12, 18, 19, 23, 30), etc., care se copiaza in banda A



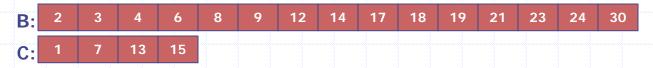
Se muta alternativ monotoniile din banda A in benzile auxiliare B si C, pana la epuizarea benzii A

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

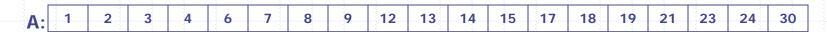
- Pentru refacerea benzii A, se vor interclasa monotoniile din benzile B si C, doua cate doua, adica:
 - (3, 8, 14, 17, 21, 24) cu (2, 4, 6, 9, 12, 18, 19, 23, 30), etc.
 - se obtin secventele (2, 3, 4, 6, 8, 9, 12, 14, 17, 18, 19, 21, 23, 24, 30), etc., care se copiaza in banda A



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



Pentru refacerea benzii A, se vor interclasa monotoniile din benzile B si C, doua cate doua, obtinandu-se:



- In acest moment, parcurgand fisierul in vederea impartirii sale pe monotonii se descopera ca el contine o singura monotonie
- Acesta este semnalul pentru oprirea algoritmului
- Sortarea prin interclasare naturala este o metoda mai buna decat sortarea prin interclasare cu 3 benzi, deoarece profita de eventualele subsecvente gata sortate ale secventei initiale
- Aceste subsecvente sunt inerente in cazul unei generari aleatoare a secventei de sortat, si apar cu o probabilitate foarte mare in practica
- In cel mai rau caz, in care secventa initiala era sortata invers, sortarea prin interclasare naturala se comporta exact la fel ca si sortarea prin interclasare cu 3 benzi