

Kurs programowania pod Oracle i Java

lista zadań nr 3

1. Dana jest tabela `dane(id, nazwa, cena)`. Przepisz wiersze tej tabeli do tabeli `dane2(id, nazwa, cena)`, ale do każdej ceny dodaj losową liczbę z przedziału $[-100, 100]$. Następnie przegladnij wszystkie wiersze tabeli `dane2` i jeśli cena w danym wierszu jest większa niż 1200, do nazwy dołącz sekwencję `'-d'`.
[1p]
2. Sprawdź, które ze sposobów odwołań do elementów tablicy indeksowej (asocjacyjnej) jest szybsze (i czy w ogóle któreś jest szybsze): poprzez bezpośrednie podawanie indeksów, czy poprzez użycie metod `FIRST` i `NEXT`. Czy sprawdzanie warunku poprzez użycie `LAST` spowalnia przeglądanie? Podaj odpowiednie przykłady na poparcie przedstawionych hipotez.
[2p]
3. Dana jest tabela `dane(id, nazwa, cena)`. Wczytaj tę tabelę do tablicy indeksowej. Zaimplementuj sortowanie `'quicksort'` (na razie nie korzystamy z procedur i funkcji) i posortuj tę tablicę wg pola `id` rekordu.
[2p]
4. Porównaj szybkość sortowania rekordów metodą z poprzedniego zadania i poprzez klauzulę `ORDER BY`.
[1p]
5. Dana jest tabela `ksiazka(sygn, autor, tytul, data_wydania)`. Wczytaj tę tabelę do następującej tabeli: indeksujemy po polu `sygn` (sygnatura ma postać *litera, cyfra, cyfra, cyfra*), a odpowiadającym elementem jest rekord o polach `autor`, `tytul` i `data_wydania` (sprawdź, czy można indeksować po typie `CHAR`). Posortuj (bąbelkowo) tabelę po polu `data_wydania`, tzn. że kolejne wywołania metody `NEXT` będą dawać rekordy zgodnie z porządkiem sortowania, a następnie sprawdź, czy istnieje 5 książek, które były wydane ciągu 3 lat.
[2p]
6. Zaimplementuj mnożenie macierzy. Utwórz trzy tablice dwuwymiarowe (czyli tablice tablic indeksowych), do dwóch pierwszych wprowadź losowe dane, a następnie do trzeciej wprowadź wynik mnożenia pierwszych dwóch. Odwołanie do elementu tablicy dwuwymiarowej: $t(i)(j)$.
[2p]

Paweł Rajba