

Zaawansowane technologie bazodanowe

Lekkie technologie bazodanowe

Ćwiczenie 1

W ćwiczeniu 1. został napisany w języku C program, który alokuje tablice 1000000 rekordów, wypełnia je danymi oraz przeszukuje zablokowane dane.

Poniżej został przedstawiony kod głównego programu:

```
int main() {
    Record** data;

    clock_t c1 = clock();
    data = generateData(SIZE);
    clock_t c2 = clock();

    if (data == NULL)
        exit(EXIT_FAILURE);

    printf("Generating data take %f seconds \n", ((float)c2-(float)c1)/CLOCKS_PER_SEC);

    int requestedId = 999999;
    c1 = clock();
    Record* foundRecord = findRecordWithId(data, SIZE, requestedId);
    c2 = clock();

    if (foundRecord == NULL)
        fprintf(stderr, "Record with id %d not found!", requestedId);
    else
        printf("Searching data take %f seconds \n", ((float)c2-
(float)c1)/CLOCKS_PER_SEC);

    displayMemoryUsage();
    freeData(data, SIZE);

    return 0;
}
```

Zwrócił on następujące rezultaty:

Generating data take 1.053422 seconds

Searching data take 0.007542 seconds

Max memory usage[kB]: 939292

Ćwiczenie 2

W ćwiczeniu drugim, należało napisać program podobny do programu z zadania pierwszego, który miał operować na bazie danych sqlite. Program miał działać w wielu konfiguracjach(użycie transakcji, użycie bazy w pliku/pamięci, różne rozmiary rekordów). Wyniki z działania programu zostały przedstawione w poniższej tabeli. Na następnej stronie znajduje się kod głównego programu.

```

int main(int argc, char* argv[]) {
    Record** data;
    data = generateData(RECORDS_COUNT);

    if (data == NULL)
        exit(EXIT_FAILURE);

    sqlite3* database = createDatabase(USE_MEMORY);

    clock_t c1 = clock();
    insertRecordsIntoDatabase(database, data, RECORDS_COUNT, USE_TRANSACTIONS);
    clock_t c2 = clock();
    printf("Inserting data take %f seconds \n", ((float)c2-(float)c1)/CLOCKS_PER_SEC);

    freeData(data, RECORDS_COUNT);

    c1 = clock();
    Record* record = findRecordInDatabase(database, REQUESTED_ID);
    c2 = clock();
    if (record == NULL) {
        fprintf(stderr, "Record with id %d not found!", REQUESTED_ID);
    } else {
        printf("Searching data take %f seconds \n", ((float)c2-
(float)c1)/CLOCKS_PER_SEC);
        free(record);
    }

    displayMemoryUsage();
    closeDatabase(database);
}

```

Tabela prezentująca wyniki:

Rozmiar rekordu	Typ bazy	Transakcje	Czas wypełniania danymi[s]	Czas przeszukiwania[s]	Zużycie pamięci[kB]
int + 20 znaków + 90 znaków	Plik	Nie	428.673096	0.024064	1005496
		Tak	3.983253	0.027138	956408
	Pamięć	Nie	5.880721	0.026973	991172
		Tak	3.676066	0.022535	944300
int + 8 znaków + 15 znaków	Plik	Nie	435.370361	0.007264	1045400
		Tak	3.293882	0.007366	1102668
	Pamięć	Nie	5.400419	0.008227	1103664
		Tak	3.244905	0.007761	1108108