

Zadania - zestaw 2

Otwarto: niedziela, 13 marca 2016, 17:45

Wymagane do: sobota, 24 marca 2018, 11:50

Zapoznaj się ze składnią i operacjami wykonywanymi przez poniższe funkcje:

- funkcje operujące na plikach i katalogach: `open`, `close`, `read`, `write`, `fcntl`, `stat`, `fstat`, `lstat`, `mkdir`, `rmdir`, `opendir`, `closedir`, `readdir`, `rewinddir`, `nftw`, `fopen`, `fclose`, `getc`, `putc`,
- funkcje i zmienne do obsługi błędów: `perror`, `errno`.

Zadanie 1. Porównanie wydajności systemowych i bibliotecznych funkcji we/wy (55%)

- (30%) Celem zadania jest napisanie programu porównującego wydajność systemowych i bibliotecznych funkcji wejścia/wyjścia. Program operować będzie na przechowywanej w pliku tablicy rekordów. Dla uproszczenia pojedynczy rekord będzie tablicą bajtów o stałej wielkości. Nazwa pliku, wielkość oraz ilość rekordów stanowią będą argumenty wywołania programu.

Program udostępniać powinien operacje:

- `generate` - tworzenie pliku z rekordami wypełnionego wygenerowaną losową zawartością (można wykorzystać wirtualny generator `/dev/random` lub w wersji uproszczonej funkcję `rand`)
- `sort` - sortuje rekordy w pliku używając sortowania przez proste wstawianie. Kluczem do sortowania niech będzie wartość pierwszego bajtu rekordu (interpretowanego jako liczba bez znaku - `unsigned char`) Podczas sortowania w pamięci powinny być przechowywane jednocześnie najwyżej dwa rekordy (sprowadza się do zamieniania miejscami i porównywania dwóch rekordów).
- `copy` - kopiuje `plik1` do `plik2`. Kopiowanie powinno odbywać się za pomocą bufora o zadanej wielkości rekordu.

Sortowanie i kopiowanie powinno być zaimplementowane w dwóch wariantach:

- `sys` - przy użyciu funkcji systemowych: `read` i `write`
- `lib` - przy użyciu funkcji biblioteki C: `fread` i `fwrite`

Rodzaj operacji oraz sposób dostępu do plików ma być wybierany na podstawie argumentu wywołania - np.:

```
./program generate dane 100 512 powinno losowo generować 100 rekordów o długości 512 bajtów do pliku dane,  
./program sort dane 100 512 sys powinien sortować rekordy w pliku dane przy użyciu funkcji systemowych, zakładając że  
zawiera on 100 rekordów wielkości 512 bajtów  
./program copy plik1 plik2 100 512 sys powinno skopiować 100 rekordów pliku 1 do pliku 2 za pomocą funkcji bibliotecznych z  
wykorzystaniem bufora 512 bajtów
```

- (25%) Dla obu wariantów implementacji przeprowadź pomiary czasu użytkownika i czasu systemowego operacji sortowania i kopiowania. Testy wykonaj dla następujących rozmiarów rekordu: 4, 512, 4096 i 8192 bajty. Dla każdego rozmiaru rekordu wykonaj dwa testy różniące się liczbą rekordów w sortowanym pliku. Liczby rekordów dobierz tak, by czas sortowania mieścił się w przedziale od kilku do kilkudziesięciu sekund. Porównując dwa warianty implementacji należy korzystać z identycznego pliku do sortowania (po wygenerowaniu, a przed sortowaniem, utwórz jego kopię). Zmierzone czasy zestaw w pliku `wyniki.txt`. Do pliku dodaj komentarz podsumowujący wnioski z testów.

Zadanie 2. Operacje na strukturze katalogów (45%)

Napisz program wyszukujący w drzewie katalogu (ścieżka do katalogu jest pierwszym argumentem programu), w zależności od wartości drugiego argumentu ('<', '>', '=') , pliki zwykle z datą modyfikacji wcześniejszą, późniejszą lub równą dacie podanej jako trzeci argument programu. Program ma wypisać na standardowe wyjście następujące informacje znalezionych plików:

- ścieżka bezwzględna pliku
- rozmiar w bajtach
- prawa dostępu do pliku (w formacie używanym przez `ls -l`, np. `r--r--rw-` oznacza prawa odczytu dla wszystkich i zapisu dla właściciela)
- datę ostatniej modyfikacji

Ścieżka podana jako argument wywołania może być względna lub bezwzględna. Program powinien wyszukiwać tylko pliki zwyczajne (bez dowiązań, urządzeń blokowych, itp). Program nie powinien podążać za dowiązaniami symbolicznymi do katalogów.

Program należy zaimplementować w dwóch wariantach:

- Korzystając z funkcji `opendir`, `readdir` oraz funkcji z rodziny `stat` (25%)
- Korzystając z funkcji `nftw` (20%)

W ramach testowania funkcji utwórz w badanej strukturze katalogów jakieś dowiązania symboliczne.