

Sprawozdanie - Projekt

<div>Piątek</div> <div>17.05</div>	<div>Systemy</div> <div>operacyjne</div>
<div>Osoby wykonujące ćwiczenie:</div> <div>Anna Brożek</div>	<div>Grupa</div> <div>nr: 1</div>

1. Opis algorytmów

a) LCFS - Last Come First Serve

Algorytm LCFS obsługuje procesy w kolejności odwrotnej do kolejności zgłoszeń. Algorytm nie wywłaszcza procesów, więc nowo przychodzący proces jest pierwszy w kolejce i czeka na zwolnienie procesora przez bieżąco wykonywany proces

b) SJF - Shortest Job First

Algorytm SJF preferuje procesy, które mają najmniejsze wymagania odnośnie czasu procesora, potrzebnego na realizację przetwarzania. W kontekście systemów komputerowych preferencje te należałoby raczej określić jako *najpierw zadanie z najkrótszą następną fazą procesora*, gdyż po odwołaniu do jądra w celu przydziału dodatkowych zasobów nastąpi zwolnienie procesora.

c) FIFO - First In First Out

Jest to rodzaj kolejki prostej, w której jak sama nazwa wskazuje, strona która pierwsza wejdzie jest pierwsza usuwana z bufora.

d) MFU - Most Frequently Used

Jest to rodzaj kolejki, w której z bufora jest usuwana najczęściej używana strona

2. Opis procedury testowania algorytmów

a) Algorytmy symulujące procesy

W plikach testowych (lcfs_test.py i sjf_test.py) znajdują się testy, które sprawdzają odpowiadające im algorytmy. Test lcfs_test.py dodatkowo generuje zadaną ilość losowych procesów i zapisuje te dane w pliku input_data.txt.

b) Kolejki

W plikach testowych (fifo_test.py i mfu_test.py) znajdują się testy, które sprawdzają odpowiadające im algorytmy. Test fifo_test.py dodatkowo generuje zadaną ilość losowych procesów i zapisuje te dane w pliku test.txt.

3. Wyniki eksperymentów

a) Algorytmy symulujące procesy

Dla algorytmów symulujących procesy wygenerowałam 10 testów z losowymi danymi wygenerowanymi przez test lcfs. Dla algorytmu lcfs średni czas oczekiwania to 16,78 a dla sjf to 8,86. Średni czas obrotu to odpowiednio: 22,6 i 14,68.

Wnioski:

- Algorytm sjf jest wyraźnie szybszy od lcfs
- W niektórych sytuacjach teoretycznych algorytm lcfs może mieć przewagę nad sjf, ale w przeważającej większości sjf jest szybszy
- W rzeczywistości algorytm lcfs nie ma zastosowania, ponieważ na procesy, które przychodzą najwcześniej czekamy najdłużej

b) Kolejki

Dla kolejek wykonałam 10 testów z losowymi danymi wygenerowanymi przez test fifo. Odpowiednio z dla fifo i mfu uzyskałam średnie błędów stron: 32 i 37,2.

Wnioski:

- Kolejka fifo jest wyraźnie szybsza niż mfu
- W niektórych sytuacjach (gdzie np. używamy najpierw jednego zestawu stron mieszczących się w buforze, a następnie kolejnego) algorytm mfu jest szybszy, ponieważ zakłada, że jeżeli jakąś stronę otwieraliśmy wiele razy, to już nam nie będzie potrzebna
- Oby dwie kolejki mogą znaleźć swoje zastosowania w rzeczywistości