

Imię i nazwisko: Piotr Paszko

Indeks: 249050

Grupa: E03-30d (Środa 13:15 – 15:00)

Przedmiot: Systemy Operacyjne - laboratorium

Temat: Symulacje planowania czasu procesora.

1. Opis programu

Program został napisany w języku python. Składa się z 3 klas oraz menu cli, klasy z których 3 implementuje obsługę procesów przez algorytmy planowania czasu procesora, zwracają one statystyki takie jak: numer, długość procesu, czas oczekiwania, etc., niezbędnych do wykonania zadania. Algorytmy planowania czasu procesora są odpowiedzialne za przydzielenie kolejności dostępu oraz zarządzanie procesom czasu procesora według określonych kryteriów.

2. Algorytmy - opis działania.

- FCFS (First Come, First Serve) – algorytm nie wywłaszczeniowy, kolejność wykonywanych procesów zależy od czasu ich nadejścia, wg zasady „pierwszy zgłoszony, pierwszy obsłużony”.
- SJF (Shortest Job First) nie wywłaszczeniowy – algorytm, w którym pierwszeństwo w realizacji ma proces o najmniejszym czasie wykonywania. Takie działanie wydajnie zmniejsza średni czas oczekiwania procesów, jednak okupione jest to możliwością głodzenia procesów o długim czasie wykonywania.
- Round Robin – algorytm wywłaszczeniowy, przydzielający pierwszemu w kolejce procesowi ustalony kwant czasu pracy procesora, po czym proces jest wywłaszczany i na jego miejsce trafia następny proces z kolejki. Po osiągnięciu końca kolejki algorytm wraca na jej początek i powtarza swoje działanie od nowa. Zaletami tego algorytmu jest brak zjawiska głodzenia procesów oraz niewystępowanie efektu konwoju.

3. Opis testowania

Uruchamiam program oraz podaje do niego dane wybrane przeze mnie („losowe”). Następnie sprawdzam poprawność tego co wykonał program przez wykonanie obliczeń na kartkach. Jeżeli obliczenia pokrywają dane przedstawione w programie to uznawałem je za poprawne.

4. Testy

```
{  
  "processes": "[ 1, 2, 3]",  
  "start_time": "[10, 5, 8]",  
  "quantum": "2",  
  "proc": "[[1, 6, 1], [2, 8, 1], [3, 7, 2], [4, 3, 3]]"  
}
```

id	start_time	burst_time	reverse_time
1	10	0	10
2	5	10	15
3	8	15	23

Average waiting time = 8.333333333333334

id	start_time	burst_time	reverse_time
1	10	13	23
2	5	10	15
3	8	13	21

Average waiting time = 12.00000

id	start_time	burst_time	reverse_time
1	6	3	9
2	8	16	24
3	7	8	15
4	3	0	3

Average waiting time = 6.75000

Na pierwszym zdjęciu widać dane losowe, które algorytmy dostają na wejście.

Długość kwantu w Round Robin ustawiam na 2.

Na zdjęciach widzimy kiedy proces zaczyna działać, jak długo zajęłoby tak jakby (wrócenie do początku) i średni czas oczekiwania na obsłużenie

```
{
  "processes": "[ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]",
  "start_time": "[10, 5, 8, 3, 6, 2, 3, 6, 7, 5]",
  "quantum": "2",
  "proc": "[[1, 6, 1], [2, 8, 1], [3, 7, 2], [4, 3, 3], [5, 6, 1], [6, 8, 1], [7, 7, 2],[8, 6, 1], [9, 8, 1], [10, 7, 2]]"
}
```

id	start_time	burst_time	reverse_time
1	10	0	10
2	5	10	15
3	8	15	23
4	3	23	26
5	6	26	32
6	2	32	34
7	3	34	37
8	6	37	43
9	7	43	50
10	5	50	55

Average waiting time = 27.0

id	start_time	burst_time	reverse_time
1	10	45	55
2	5	34	39
3	8	44	52
4	3	24	27
5	6	37	43
6	2	10	12
7	3	27	30
8	6	39	45
9	7	46	53
10	5	43	48

Average waiting time = 34.90000

id	start_time	burst_time	reverse_time
1	6	3	9
2	8	42	50
3	7	20	27
4	3	0	3
5	6	9	15
6	8	50	58
7	7	27	34
8	6	15	21
9	8	58	66
10	7	34	41

Average waiting time = 25.80000

Druga seria badawcza z większą ilością danych.

5. Wnioski

Zgodnie z oczekiwaniami SJF, osiągnął najkrótsze średnie czasy oczekiwania.

Natomiast najdłuższe Round Robin.

FCFS osiągnął czasy uplasowanie pomiędzy tymi dwoma.

Można By wysunąć wniosek, że SJF jest najlepszym algorytmem w tym zestawieniu.

Jednakże należy nie zapominać, że może być to okupione głodzeniem procesów.

Czego możemy uniknąć używając Round Robin.

Dlatego wybór algorytmu zawsze powinien być na podstawie tego do obsługi jakiego typu procesów zostanie on wykorzystany.