

Teoria współbieżności

Problem pięciu filozofów

Łukasz Stępień

1. Temat ćwiczenia.

Porównanie różnych rozwiązań problemu pięciu filozofów i ich implementacji. Zaprojektowanie algorytmów jednoczesnej alokacji współdzielonych zasobów (widelce) przez konkurujące procesy (filozofowie), tak aby uniknąć zakleszczenia i zagłodzenia. Warianty:

1. **Rozwiązanie naiwne (z możliwością blokady).** Każdy filozof czeka, aż wolny będzie lewy widelec, a następnie go podnosi (zajmuje), następnie podobnie postępuje z prawym widelcem.
2. **Rozwiązanie z możliwością zagłodzenia.** Każdy filozof sprawdza czy oba sąsiednie widelce są wolne i dopiero wtedy zajmuje je jednocześnie. Rozwiązanie to jest wolne od blokady, jednak w przypadku, gdy zawsze któryś z sąsiadów będzie zajęty jedzeniem, nastąpi zagłodzenie, gdyż oba widelce nigdy nie będą wolne.
3. **Rozwiązanie asymetryczne.** Filozofowie są ponumerowani. Filozof z parzystym numerem najpierw podnosi prawy widelec, filozof z nieparzystym numerem najpierw podnosi lewy widelec.
4. **Rozwiązanie stochastyczne.** Każdy filozof rzuca monetą tuż przed podniesieniem widelców i w ten sposób decyduje, który najpierw podnieść - lewy czy prawy (z prawdopodobieństwem 1 nie dojdzie do zagłodzenia [3]).
5. **Rozwiązanie z arbitrem.** Zewnętrzny arbiter (lokaj, kelner) pilnuje, aby jednocześnie co najwyżej czterech (w ogólnym przypadku $N-1$) filozofów konkurowało o widelce. Każdy podnosi najpierw lewy a potem prawy widelec. Jeśli naraz wszyscy filozofowie będą chcieli jeść, arbiter powstrzymuje jednego z nich aż do czasu, gdy któryś z filozofów skończy jeść.
6. **Rozwiązanie z jadalnią.** Rozwiązanie jest modyfikacją wersji z arbitrem. Filozof, który nie zmieści się w jadalni (czyli arbiter nie pozwolił mu jeść) je „na korytarzu” podnosząc jednorazowo widelce w odwrotnej kolejności (do reszty filozofów w jadalni).

2. Implementacja.

Każdy z powyższych wariantów został zaimplementowany i przetestowany w środowisku Javy. Wykresy zostały stworzone za pomocą biblioteki *matplotlib* w środowisku Python. Cały kod jest dostępny w załączniku do tego sprawozdania.

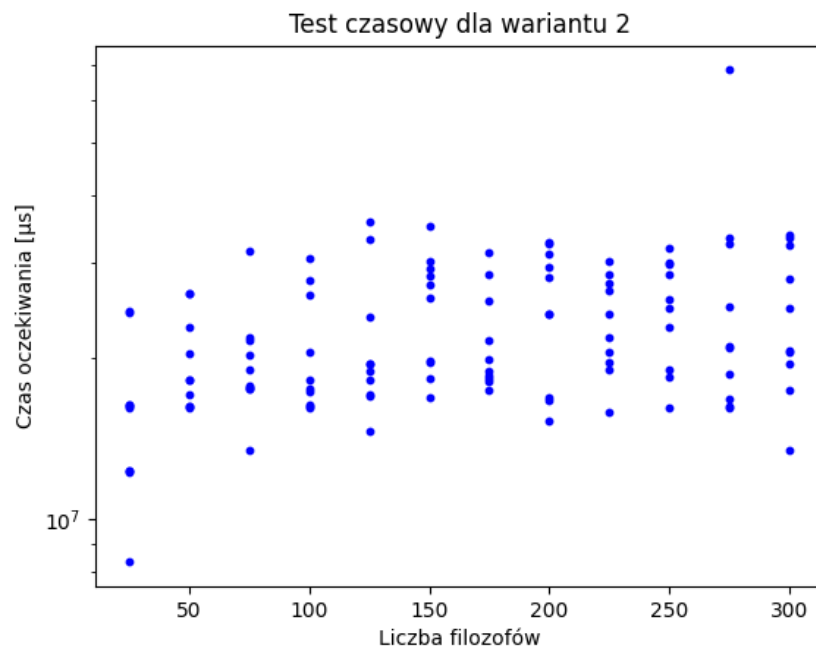
Testy czasowe zostały zaimplementowane w następujący sposób:

- Każdy wątek sumuje czasy oczekiwania na oba zasoby oraz liczbę udanych prób osiągnięcia ich
- Po zakończeniu wszystkich wątków metoda *Test* sumuje średnie czasy oczekiwania każdego wątku
- Finalnie metoda zwraca średnią arytmetyczną czasu oczekiwania dla wszystkich wątków
- W klasie *Main* procedura *Test* jest uruchamiana dla danej ilości filozofów dziesięć razy, a wyniki zapisywane są do pliku *result[n].csv*, który następnie jest wczytywany przez program *.py* tworzący odpowiednie wykresy.

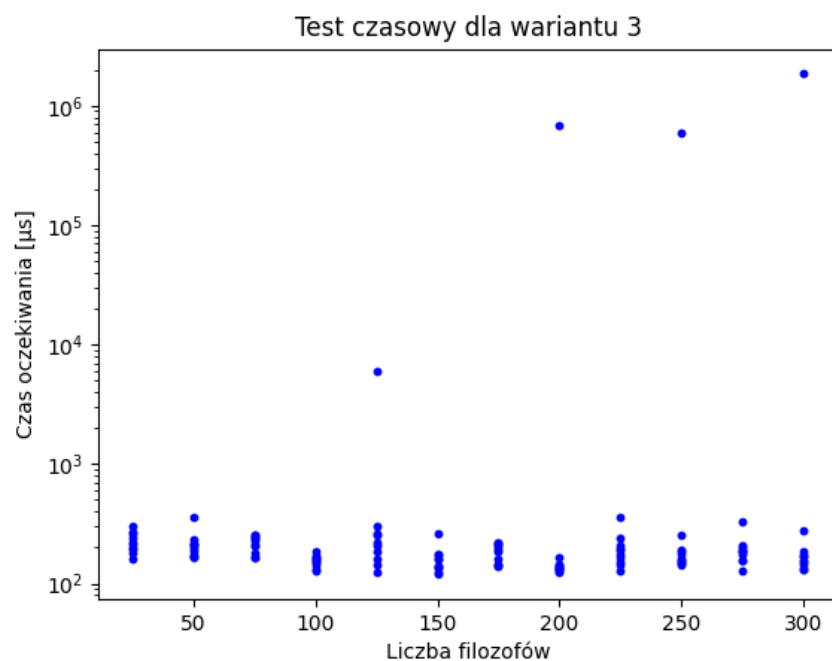
3. Wyniki.

Wykresy przedstawiają rozkład średnich czasów oczekiwania w zależności od ilości filozofów. Dla każdej ilości filozofów zamieszczono po dziesięć wyników testów.

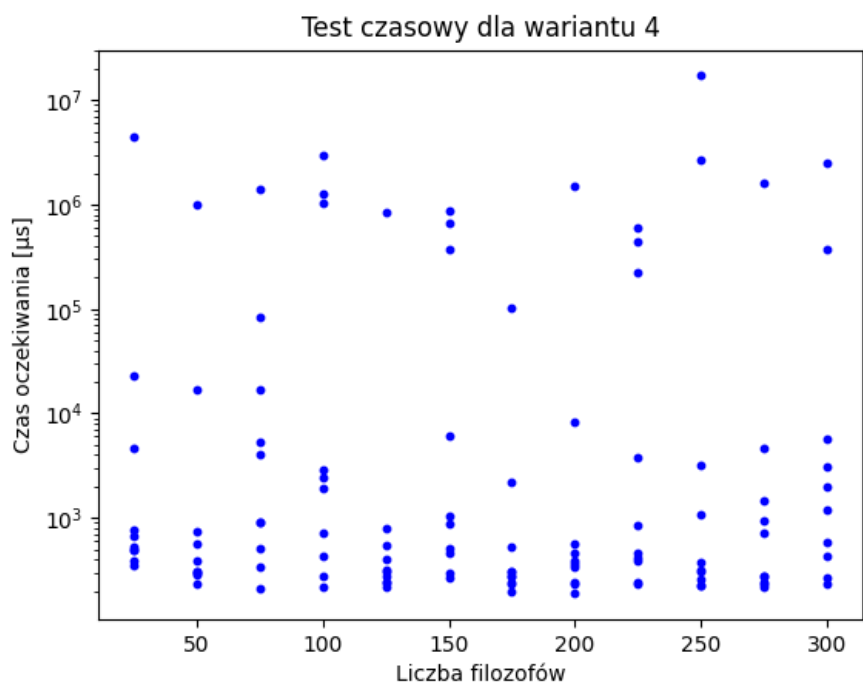
1. **Rozwiązanie naiwne:** po pewnym czasie działanie program zakleszcza się, brak testów czasowych.
2. **Rozwiązanie z możliwością zagłodzenia:**



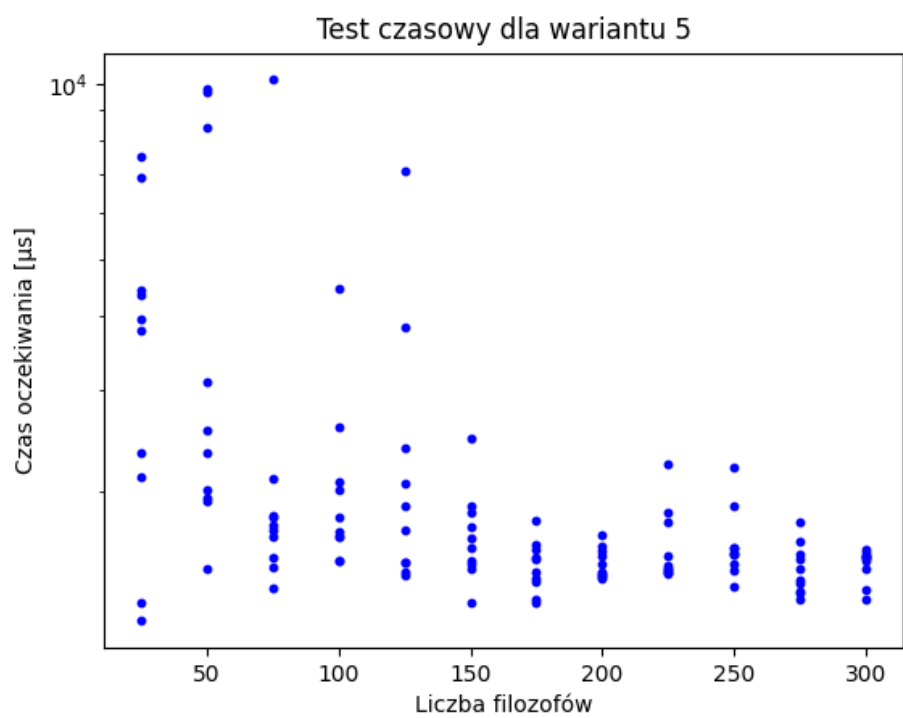
3. **Rozwiązanie asymetryczne:**



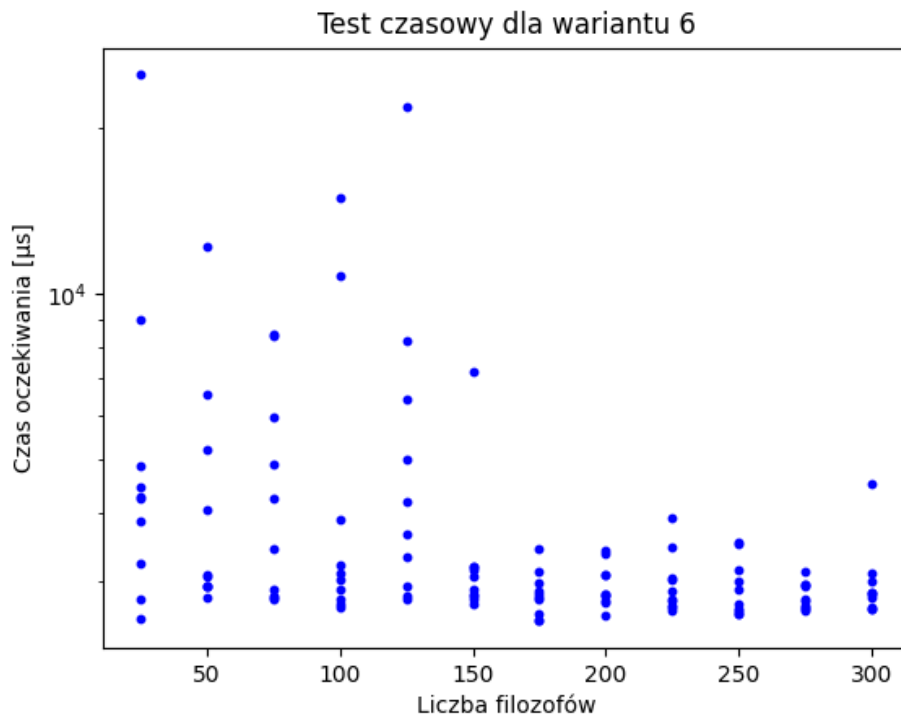
4. Rozwiązanie stochastyczne:



5. Rozwiązanie z arbitrem:



6. Rozwiązanie z jadalnią:



4. Wnioski.

1. **Rozwiązanie naiwne:** zgodnie z przewidywaniami dochodzi do zakleszczenia
2. **Rozwiązanie z możliwością zagłodzenia:** bardzo wysokie średnie czasy oczekiwania na zasoby może oznaczać małą efektywność tego wariantu.
3. **Rozwiązanie asymetryczne:** czas oczekiwania na zasoby jest bardzo niski, tylko w czterech przypadkach średnia jest znacząco zawyżona, co może oznaczać długie oczekiwanie któregoś z zasobów, przeważająca liczba średnich oczekiwań jest skoncentrowana w tym samym przedziale czasowym, co może wskazywać na pewną stabilność tego podejścia, brak wyraźnej zmiany w stosunku do ilości filozofów
4. **Rozwiązanie stochastyczne:** wyniki są znacząco rozproszone na szerokim przedziale czasowym, co może wskazywać na wiele długich czasów oczekiwań na zasoby, nie są one zależne od ilości filozofów
5. **Rozwiązanie z arbitrem:** wyniki dla ilości filozofów 25-125 są widocznie bardziej rozproszone niż w przedziale 150-300, z czego można wnioskować o lepszym działaniu tego wariantu dla większej liczby filozofów
6. **Rozwiązanie z jadalnią:** wyniki są znacząco podobne do wariantu piątego, z czego można wnioskować, że subtelna różnica w logice tych wariantów ma małe znaczenie

Średnie czasy oczekiwania są znacząco wyższe dla wariantu z możliwością zagłodzenia, co widoczne jest w wariancie drugim. Brak mechanizmów synchronizacji zwiększa czas oczekiwania na dostęp do zasobów, np. dla wariantu 1. zachodzi zakleszczenie.

5. Bibliografia.

- https://pl.wikipedia.org/wiki/Problem_ucztujących_filozofów