Podaj definicję wątku (ang. thread).

https://pl.wikipedia.org/wiki/W%C4%85tek\_(informatyka)

Jakie korzyści daje zastosowanie wątków? (4.4, 4.7)

Jakie niezależne zasoby posiada każdy wątek? (4.5)

Jakie zasoby współdzielone są między wątkami w obrębie procesu? (4.5)

Podaj przykład wielowątkowej architektury serwera. (4.6)

Jakie wyzwania wiążą się z programowaniem wieloprocesorowym? (4.8)

Jaka jest różnica między równoległością (ang. Parallelism) a współbieżnością (ang. Concurrency)? (4.8, 4.9)

Jakie mamy typu obliczeń równoległych? (4.10, 4.11)

- podaj po jednym przykładzie dla każdego typu

Omów wątki jądra i wątki użytkownika. (4.14, 4.15)

Omów wielowątkowy model many-to-one. (4.17)

Omów wielowątkowy model one-to-one. (4.18)

Omów wielowątkowy model many-to-many. (4.19)

Omów wielowątkowy model two-level-model. (4.20)

Wymień dwa sposoby implementacji API do wielowątkowości. (4.21)

Omów specyfikację pthreads wątków POSIX. (4.22)

Omów wątki Javy. (4.28)

Podaj przykład implementacji wątku Java. (4.29)

Omów wielowątkowość niejawną (ang. implicit threading). (4.33)

Omów zachowanie wywołań systemowych fork() oraz exec() dla procesów wielowątkowych. (4.50)

- w jakim przypadku powielamy wszystkie wątki?

- w jakim przypadku powielamy tylko wątek wywołujący funkcję fork?

Omów obsługę sygnałów (ang. signal handling) w systemach unixowych. (4.51, 4.52)

- przykłady w pliku signals.docx i katalogu signals

Omów kasowanie wątków (ang. thread cancellation). (4.53, 4.54)

- podać przykład, kiedy zachodzi potrzeba skasowania wątku

Omów pamięć lokalną wątku. (ang. Thread-Local Storage) (4.56)

Omów wątki Linux’a i wymień flagi kontrolne funkcji clone. (4.62)