

Лабораторна робота №6 (семестр 2)

ТЕМА: Системна взаємодія між процесами.

МЕТА: Отримати практичні навички в використанні функцій Win API. Навчитися організовувати коректну взаємодію між процесами застосовуючи механізм:

– Створення потоку у сторонньому процесі.

МЕТА: Отримати практичні навички використання функцій Win API.

ЗНАТИ: Основи програмування в ОС Windows.

ВМІТИ: Застосовувати API-функції.

ТЕОРЕТИЧНІ ВІДОМОСТІ.

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

ЗАВДАННЯ

Використовуючи наявну електронну документацію повністю повторити функціонал програм наведених в прикладах на мові C++ чи C#.

Тобто провести перетворення програм (трансляцію), представлених на мові програмування Object Pascal на мову C++ чи C#.

ПРИКЛАДИ

"6. RemoteThread" – Системна взаємодія між процесами через створення потоку у сторонньому процесі.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Для чого використовують механізми IPC?
2. Які способи міжпроцесної взаємодії ви знаєте?
3. Який спосіб IPC є найпростішим?
4. За допомогою якого повідомлення можна реалізувати передачу даних між процесами?
5. Чому для надсилання даних іншому процесу не можна використовувати функцію *PostMessage*?
6. Опишіть послідовність дій при взаємодії між процесами через *WM_COPYDATA*.
7. Напишіть фрагмент програми для передачі структури даних *TData* вікну з відомим дескриптором за допомогою *WM_COPYDATA*.
8. У чому полягає відображення файлу на пам'ять?
9. Для чого використовують механізм відображення файлів?
10. Чи можна реалізувати IPC за допомогою відображених файлів для процесів на різних машинах мережі?
11. Опишіть послідовність дій для реалізації IPC за допомогою відображення файлів.
12. Як відбувається створення об'єкта "файл"?
13. Як дисковий файл відображується у пам'ять?
14. Для чого використовується об'єкт ядра "канал"?
15. Які бувають канали?
16. Вкажіть послідовність дій для міжпроцесної взаємодії за допомогою іменованих каналів.
17. Який формат назви іменованого каналу?
18. Який процес є сервером каналу?
19. Чи можна реалізувати IPC для процесів на різних машинах за допомогою каналів?
20. Чому для роботи з каналами використовують ті ж функції, що і для роботи з файлами?
21. Які ключові особливості поштових скриньок?
22. Вкажіть формат назви поштової скриньки.
23. Опишіть послідовність дій для здійснення IPC за допомогою поштової скриньки.
24. Чи можна реалізувати передачу даних процесу на іншому комп'ютері за допомогою поштової скриньки?
25. Напишіть фрагмент програми для передачі даних іншому процесу через поштову скриньку *PostBox*.
26. Напишіть фрагмент програми для отримання даних від іншого процесу через поштову скриньку *PostBox*.
27. Як реалізувати двонаправлену передачу даних через поштову скриньку?
28. Як передати повідомлення поштовим скринькам відразу на кількох комп'ютерах?
29. Що таке "сокет"?
30. У чому відмінність використання сокетів від інших механізмів IPC?
31. Як ідентифікується сокет?
32. Опишіть послідовність дій для передачі даних через сокет.
33. Для чого використовується порт сокета?
34. У чому різниця між синхронними та асинхронними сокетами?
35. Назвіть головні функції *Winsock API* для передачі даних через сокет.
36. Для чого використовують функцію *socket*?
37. Намалюйте блок-схему багатопотокового сервера, який отримує дані від клієнтів через сокет.
38. Для чого використовують функцію *Winsock API listen*?
39. Чи можна з одного процесу створити потік в іншому процесі?
40. Вкажіть послідовність дій, за допомогою яких можна завантажити сторонній код в адресний простір іншого процесу.