

Proszę czekać...

Trwa prezentacja wykładu...

Dawid Najgiebauer

O czym?

4% 2 z 22

Pozostały czas: 70 min

Kiedy użytkownik się zirytuje...

Krótko o psychologii używania programów oraz jak system potrafi pomóc i przeszkodzić

...i jak go uspokoić?

Kilka propozycji informowania o wykonywaniu poleconych zadań

Jak wpłynie to na program i jak ten wpływ zminimalizować?

O zadaniach w wątku głównym i komunikacji między wątkami

Kiedy użytkownik się zirytuje...

9% 3 z 22

Pozostały czas: 66 min

- Ludzkie oko jest w stanie rejestrować ok. 200 zmiennych obrazów na sekundę – zmiany w czasie poniżej 5 ms są niedostrzegalne*
 - * Podczas wyświetlania obrazu najczęściej większe znaczenie ma częstotliwość odświeżania wyświetlacza
- Czas reakcji człowieka wynosi ok. 250 ms
- Czas reakcji programu na poziomie powyżej $\frac{1}{4}$ sek. zaczyna być postrzegany jako nienaturalny, wydłużony

Kiedy użytkownik się zirytuje...

13% 4 z 22

Pozostały czas: 63 min



Kiedy system Windows się zirytuje...

18% 5 z 22

Pozostały czas: 60 min

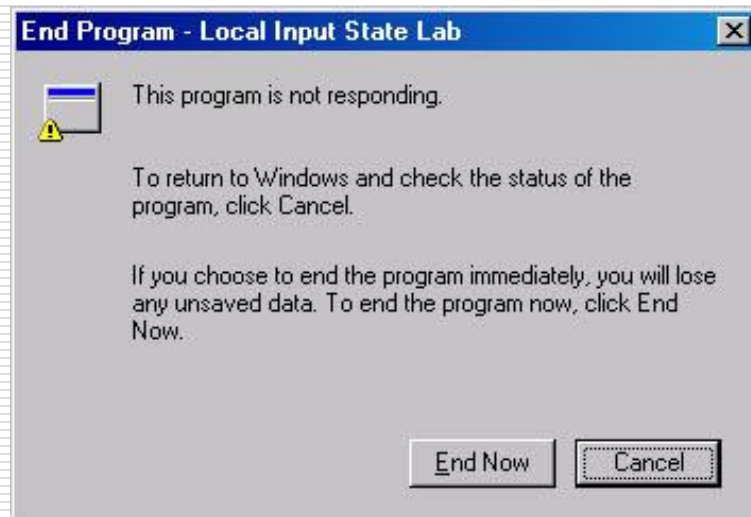
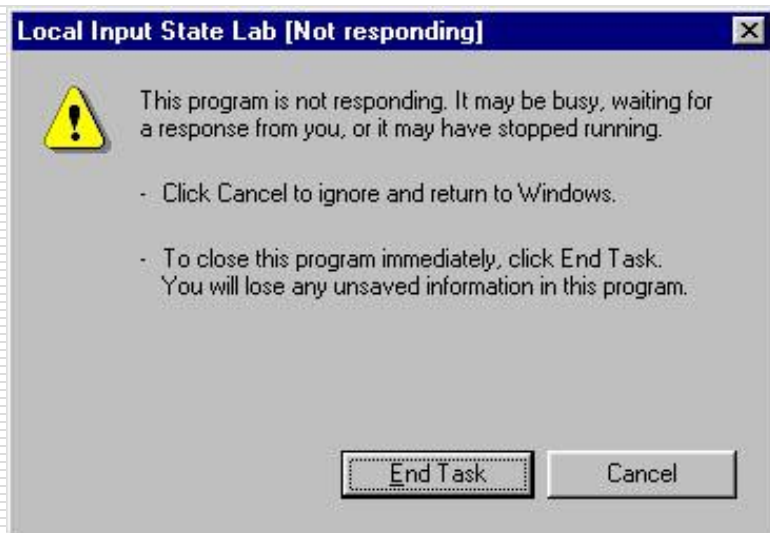
- System Windows wykrywa zawieszona aplikacje przez sprawdzenie udzielania odpowiedzi na komunikaty z kolejki
- System nie sprawdza sam z siebie aplikacji (może się to stać tylko po wysłaniu do niej wiadomości)
- Limit czasu wynosi 5 sekund
- Aplikacje bez interfejsu, które nie otrzymują komunikatów interfejsu, mogą pozostać zawieszona bez wykrycia

Kiedy system Windows się zirytuje...

23% 6 z 22

Pozostały czas: 57 min

❑ Windows 98 oraz próba zamknięcia w 2000/XP

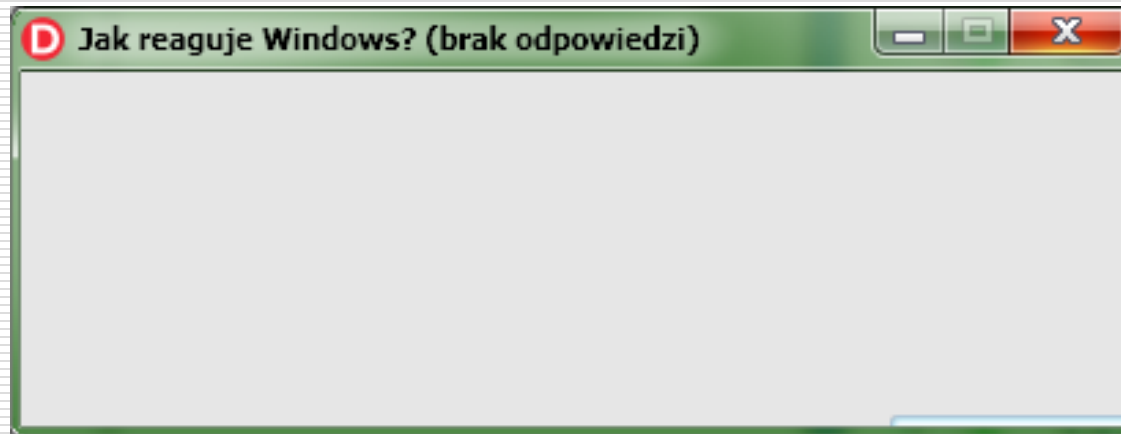


Kiedy system Windows się zirytuje...

27% 7 z 22

Pozostały czas: 53 min

- W Windows 2k+ wcześniej okno aplikacji jest ukrywane i zastępowane oknem-widmem – okno ma kursor klepsydry oraz zmieniony nagłówek – reaguje na przesuwanie

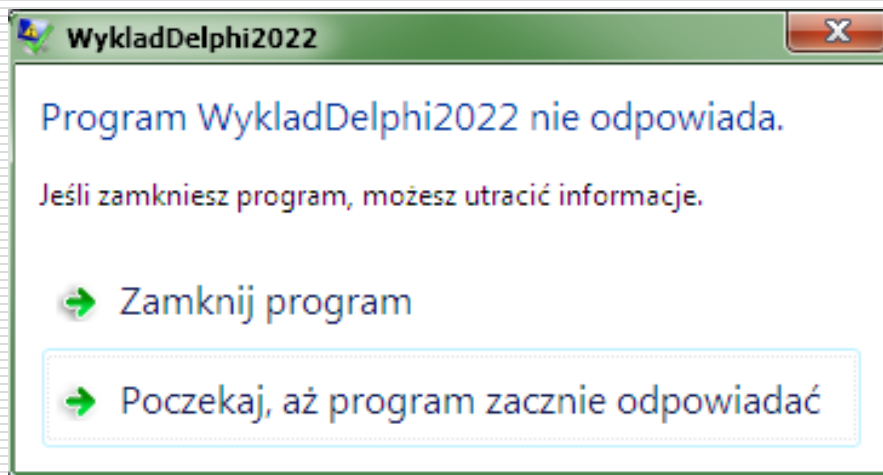


Kiedy system Windows się zirytuje...

32% 8 z 22

Pozostały czas: 50 min

- ❑ W przypadku próby zamknięcia nieodpowiadającego okna wyświetlany jest komunikat



Wątek główny czy dodatkowy?

36% 9 z 22

Pozostały czas: 46 min

- Bezpieczny limit czasu operacji w wątku głównym: poniżej 3 sek.
 - Bezpieczny czas wynika z reakcji systemu na brak obsługi komunikatów
- Wątek dla operacji trwających <250 ms?
 - Utworzenie, uruchomienie i obsługa wątku kosztuje czas
 - `TTask` (i pokrewne) dla optymalizacji nie niszczą wątków przeznaczonych na operacje
 - Jeśli użytkownik spodziewa się, że operacja zajmie czas, jego tolerancja na reakcję programu wzrasta
- Zajęcie wątku głównego uniemożliwi wyświetlanie animacji z komponentu `TAnimate!` (n/d starych wer.)

Gdy wątek aplikacji pracuje...

41% 10 z 22

Pozostały czas: 43 min

- Przełączenie właściwości `Enabled` wszystkich widocznych kontrolek
(uwaga z przywracaniem stanu!)
- Przełączenie właściwości `Enabled` całego okna (?)
- Otwarcie okna modalnego
- Jeśli okno włączone, konieczna obsługa zdarzenia `OnCloseQuery` lub `OnClose` uniemożliwiającego zamknięcie okna

Gdy zadania są na wątku głównym

45% 11 z 22

Pozostały czas: 40 min

- ❑ Choć nie powinno mieć to miejsca, to czasem czasochłonne zadania wykonywane są bezpośrednio na komponentach VCL, co zmuszałoby do wstawienia większości kodu w metodę `Synchronize` – to pozbawia sensu użycie wątku
- ❑ Metoda `Application.ProcessMessages` pozwala na przetworzenie aktualnej kolejki komunikatów i powstrzymuje system przed sugerowaniem zawieszenia się programu
- ❑ Zbyt częste wywołanie w/w metody ograniczy wydajność; zbyt rzadkie – nie przyniesie zamierzonego rezultatu. Sprawdzi się tylko tam, gdzie częstość wywołań jest przewidywalna lub możliwa do określenia

Komunikacja wątku z interfejsem

50% 12 z 22

Pozostały czas: 36 min

- ❑ Metoda `Synchronize` pozwala odwoływać się do komponentów VCL z poziomu wątku. Blokuje dalszą pracę do momentu wykonania zadań przez wątek główny.
- ❑ Gdy istnieje duże prawdopodobieństwo, że kolejne zadania dla wątku głównego zdążą się wykonać podczas obliczeń, a nie jest istotne zapewnianie precyzyjnej informacji, możliwe jest zlecenie zadania bez czekania na wykonanie przy użyciu metody `Queue`.
- ❑ Alternatywą dla `Synchronize` i `Queue` może być przesłanie komunikatu do okna przez, odpowiednio, `SendMessage` i `PostMessage`.

Jak często odświeżać informacje?

54% 13 z 22

Pozostały czas: 33 min

- W przypadku, gdy pojedyncze zadanie jest długotrwałe pozostawienie stałego opisu przestaje się sprawdzać. Przykładem pobieranie pliku, podczas którego można pokazywać aktualną szybkość i pozostały czas.
- Zbyt częste odświeżanie (<0,5 sek. dla informacji tekstowych) uniemożliwia użytkownikowi odczyt i staje się mniej użyteczne, jednocześnie zwiększając obciążenie wątku VCL.
- O potrzebie odświeżenia wyświetlanej informacji może decydować wątek główny

Jak zdeterminować częstość?

59% 14 z 22

Pozostały czas: 29 min

- Bezwarunkowe odświeżenie informacji tylko, gdy atomowa operacja trwa długo
- Dla dużych liczby przebiegów pętli, odświeżanie co któryś przebieg dający pożądaną częstość. Aby w stopniu minimalnym wpłynąć na wydajność, lepiej użyć operatora `and` niż często używanego `mod`.
- Funkcja `GetTickCount` nie powinna stanowić podstawy w określeniu częstości odświeżania.

Jak zdeterminować częstość? – cd.

64% 15 z 22

Pozostały czas: 26 min

- Wątek może wysyłać wszystko, ale o odświeżeniu decyduje główny – w ten sposób wątek nie traci czasu na interfejs, a informacje są aktualne
- Zamiast wysyłania przez wątek informacji o postępie zawsze lub z określonym interwałem, można od wątku zażądać takiej informacji co określony z góry czas.

Pomysł 1. Cursor

68% 16 z 22

Pozostały czas: 22 min

- Prosty w wykonaniu
- Nie potrzebuje osobnego wątku ani przetworzenia kolejki komunikatów
- Narzut dla czasu operacji jest pomijalny
- Cursor klepsydry zwiększa tolerancję na brak reakcji ze strony aplikacji
- Ciekawa animacja może jeszcze bardziej wydłużyć czas tolerancji
- Jednorodność operacji nie ma znaczenia
- Dla operacji poniżej czasu 250 ms, przełączający się cursor może irytować

Pomysł 2. Okno modalne

73% 17 z 22

Pozostały czas: 19 min

- Przejmuje obsługę komunikatów, zatem poprawne zaprojektowanie jest znacznie prostsze
- Umożliwia wyświetlenie dodatkowych informacji w razie potrzeby
- Użycie okna dla krótkotrwałych operacji (czas wyświetlania poniżej czasu potrzebnego na odczyt zawartości) może być irytujące

Pomysł 3. Pasek postępu

77% 18 z 22

Pozostały czas: 16 min

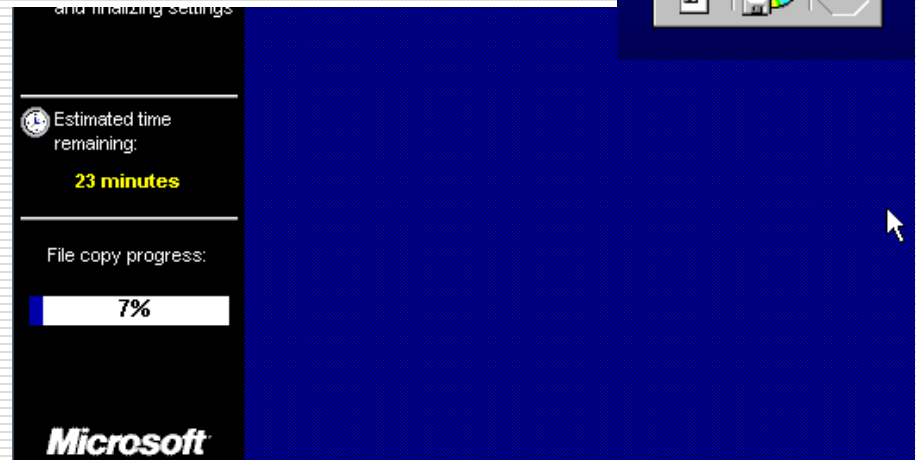
- Nada się tam, gdzie z góry znamy całkowitą ilość operacji
- Najlepiej sprawdza się do operacji, której cząstkowy czas jest stały
- Nada się dla operacji trwających od kilkunastu sekund do kilkunastu minut
- Standardowy komponent jest bezpieczny wątkowo!

Pomysł 3. Pasek postępu – wkurza, gdy...

82% 19 z 22

Pozostały czas: 12 min

- Dojdzie do końca i znów zaczyna od początku
- Nie dojdzie do końca, a już po wszystkim
- Działa bardzo skokowo
- Nagle zwalnia lub przyspiesza
- Przesuwa się o 1 piksel na kilka minut



Pomysł 4. Okno z listą etapów

88% 20 z 22

Pozostały czas: 9 min

- Sprawdza się, gdzie w ramach operacji wykonywane są zróżnicowane, ale definiowalne czynności
- W przypadku błędu daje informację o miejscu wystąpienia
- Dobrze uwidacznia ilość pracy, jaka została wykonana i jaka czeka – użytkownik ma bardzo wysoki poziom tolerancji na czas trwania
- Nadaje się także tam, gdzie część z operacji jest szybka, ale może powodować błąd

Pomysł 5. Wyświetlanie częściowych wyników

94% 21 z 22

Pozostały czas: **5 min**

- Gdy w ramach operacji zwracanych jest wiele wyników
- Nie jest znana z góry ich liczba (szczególnie nadaje się do pobierania wyników z baz danych lub strumieni)
- Gdy wyświetlenie części wyników może dać istotną informację
- Czas odświeżania od 0,2 do 1 sek.

Podsumowanie pomysłów

99% 22 z 22

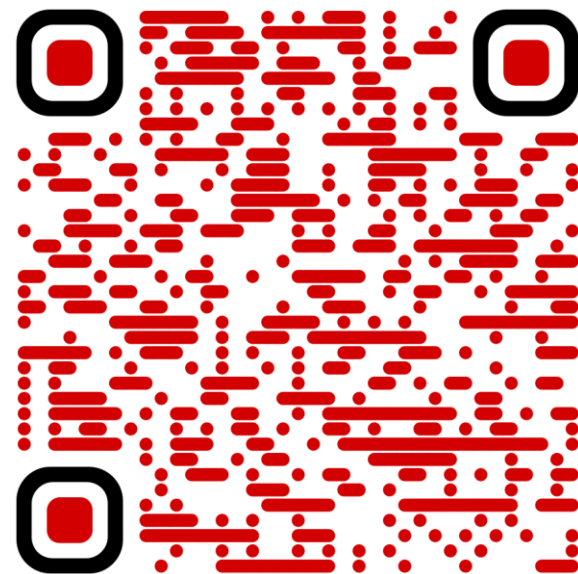
Pozostały czas: 2 min

Kursor	<input type="checkbox"/> Zawsze dla operacji >250 ms
Proste okno	<input type="checkbox"/> Dla operacji od kilku do kilkadziesiątu sekund <input type="checkbox"/> Możliwość wzbogacenia o opis aktualnej czynności, co umożliwia użycie do operacji trwających kilka minut
Pasek postępu	<input type="checkbox"/> Od kilkunastu sekund do kilkunastu minut <input type="checkbox"/> Głównie dla operacji o jednorodnym czasie postępu
Okno z listą etapów	<input type="checkbox"/> Gdy łączny czas przekracza kilka sekund, a operacje nie są jednorodne <input type="checkbox"/> Gdy informacja o etapie może być istotna
Wyświetlanie części wyników	<input type="checkbox"/> Gdy łączny czas jest nieprzewidywalny <input type="checkbox"/> Gdy chcemy pokazać szczegółowe informacje o już dostępnych wynikach

Wykład został zakończony!

Zaprezentowane informacje i więcej na stronie:

www.dzyszla.pl/porada-21.html



Index (24) out of range

104% 24 z 22

Pozostały czas: -5 min

- Różnica między pętlami `while`, a `repeat..until`

