

ZASADY ZALICZENIA PRZEDMIOTU SYSTEMY ODPORNE NA BŁĘDY PROJEKT

mgr inż. Michał Młodawski

Katedra Systemów Informatycznych PŚk

1. Obecność na zajęciach projektowych jest obowiązkowa.
2. Od studenta oczekuje się pracy nad projektem przez cały semestr i systematyczne przedstawianie postępów na zajęciach projektowych prowadzącemu.
3. Oceniane będą następujące elementy:
 - a. Sprawozdanie, które powinno zawierać elementy takie jak:
 - I. Temat pracy,
 - II. Nazwiska autorów,
 - III. Opis użytych technologii,
 - IV. Opis zastosowanych algorytmów,
 - V. Diagramy głównych klas,
 - VI. Diagramy przypadków użycia,
 - VII. Przedstawienie działania aplikacji w formie zrzutów ekranu,
 - VIII. Wnioski
 - b. Działanie programu i spełnienie założeń projektowych,
 - c. Znajomość projektu
4. Łączna liczba punktów sumuje się do 100. Ocena końcowa zależy od łącznej liczby uzyskanych punktów:
 - a. 0–49 punktów - 2,0 niedostateczny,
 - b. 50–59 punktów - 3,0 dostateczny,
 - c. 60–69 punktów - 3,5 dostateczny plus,
 - d. 70–79 punktów - 4,0 dobry,
 - e. 80–89 punktów - 4,5 dobry plus,
 - f. 90–100 punktów - 5,0 bardzo dobry
5. Link do repozytorium z plikami projektu i historią zmian na e-mail prowadzącego dydaktyka@pm.me z tematem **S_SOnB_PO_NUMER-TEMATU_GRUPA_NAZWISKO**.
6. **W razie wykrycia plagiatu praca zostanie oceniona na zero punktów bez możliwości poprawy.**
7. Student przed rozpoczęciem kursu ma obowiązek zapoznać się z instrukcją BHP obowiązującą w laboratorium i poświadczyć to na piśmie.
8. Dopuszczalne jest odstępianie od niniejszych reguł w uzasadnionych przypadkach (zaświadczenia o niepełnosprawności, okres rekonwalescencji, itp.), tylko za obustronnym porozumieniem osób zainteresowanych i prowadzącego. Jednakowoż

niedopuszczalne jest ograniczenie zakresu materiału, którego opanowanie w ramach zajęć określone jest w karcie przedmiotu.

9. W sprawach nieuregulowanych niniejszymi zasadami zastosowanie mają ogólnie obowiązujące przepisy, w szczególności Regulamin Studiów i ogólnie przyjęte zasady.

ORGANIZACJA ZAJĘĆ PRZEDMIOTU SYSTEMY ODPORNE NA BŁĘDY

1. Wychodzenie z sali dozwolone jest tylko po zgłoszeniu prowadzącemu.
2. Na laboratoriach obowiązuje zakaz spożywania posiłków i napojów.
3. Okrycia wierzchnie należy pozostawić w szatni.
4. Plecaki, czy torby należy umieścić w miejscu nie przeszkadzającym w chodzeniu.
5. Podczas laboratorium należy zachować ciszę. Konwersację z innymi członkami zespołu należy prowadzić, tak aby nie zakłócić ciszy
6. Na każdych zajęciach sprawdzana jest lista obecności.
7. Instrukcje laboratoryjne umieszczane są pod adresem: achilles.tu.kielce.pl lub github.com/SimpleMethod/PSK-SemestZimowy
8. Literatura podstawowa i uzupełniająca podana jest w karcie przedmiotu

WYMAGANIA PROJEKTOWE PRZEDMIOTU SYSTEMY ODPORNE NA BŁĘDY

1. Projekt powinien wykorzystywać system kontroli wersji Git i być umieszczony w repozytorium do którego prowadzący ma dostęp,
2. Dopuszcza się wykorzystanie dowolnego języka programowania,
3. Komunikacja powinna być oparta o TCP/IP lub pamięci dzielonej z wykorzystaniem semaforów lub innym rozwiązaniem umożliwiającym komunikację w czasie rzeczywistym pomiędzy elementami systemu,
4. Każdy element systemu powinien działać jako osobny wątek, proces, program,
5. Celem projektu jest wierne zasymulowanie działania algorytmu, a nie jego implementacja. Niedopuszczalnym jest użycie gotowych bibliotek implementujących rozwiązanie lub jego część.
6. Program powinien posiadać interfejs graficzny umożliwiający:
 - a. Monitorowanie pracy poszczególnych elementów systemu,
 - b. Możliwość wprowadzania usterki i jej usuwania w poszczególnych elementach systemu,
 - c. Możliwość wprowadzania nowych wartości do systemu
7. **Program powinien mieć funkcjonalność wstrzykiwania minimum 3 różnych rodzajów błędów do każdego elementu systemu, nawet do elementów, które nie tolerują danego rodzaju błędu.**

TEMATY PROJEKTÓW 2021/2022

Numer	Temat	Opis
1	Paxos głosowanie	Zastosowanie algorytmu Paxos do głosowania nad wartością. System powinien składać się z 8 serwerów z zdefiniowanym liderem, algorytm ma na celu porozumiewanie się pomiędzy elementami systemu i ustalenie na drodze głosowania wspólnej wartości, a następnie jej zwrócenie.
2	Paxos wybór lidera	Algorytm paxos wybiera lidera spośród dostępnych serwerów i aktualizuje wartość. System powinien składać się z 6 serwerów. Należy zapewnić możliwość wykrycia uszkodzonego lidera i wybór nowego.
3	RAID	Symulacja macierzy dyskowych typu RAID0, RAID1, RAID3 złożonych z 4 symulowanych dysków, zawierających co najmniej 128 sektorów o pojemności co najmniej 32 bajtów każdy. Symulacja pracy pod obciążeniem (losowym) i tworzenie statystyk obciążenia poszczególnych dysków.
4	CRC	Symulacja pracy 10 komputerów połączonych ze sobą w kształt grafu. Pomiędzy wybranymi przez użytkownika węzłami można przysyłać informacje zabezpieczone kodem CRC o długości 16 bitów (dowolny wielomian wprowadzany przez użytkownika). Symulacja błędów wielokrotnych.
5	Głosowanie przybliżone	Symulacja pracy 8 serwerów czasu (co najmniej godzina, minuta i sekunda) połączonych w topologii gwiazdy z centralnym komputerem, wyznaczającym w drodze głosowania przybliżonego o najbardziej prawdopodobny czas. Możliwość ustawienia wag (przez użytkownika) dla każdego komputera satelitarnego z osobna.
6	Wykrywanie błędów kodem Bergera	System składający się z 8 serwerów połączonych w graf. Serwer nadzorujący wysyła 16 bitową informację zabezpieczoną kodem korekcyjnym Bergera.
7	Nadmiarowość TMR	Symulacja działania pamięci dzielonej w układzie TMR. Symulacja oparta o 6 serwerach
8	Nadmiarowość TMR	Symulacja wykonywania obliczeń w układzie TMR. Symulacja oparta o 6 serwerach
9	Zatwierdzenie dwufazowe	System powinien składać się z 6 serwerów oraz koordynatora. System powinien aktualizować wartość i ją zwracać.
10	Korekcja błędów Hamminga	System składający się z 8 serwerów połączonych w graf. Serwer nadzorujący wysyła 16 bitową informację zabezpieczoną kodem korekcyjnym Hamminga.
11	Korekcji błędów Reeda-Solomona	System składający się z 4 serwerów połączonych w graf. Lider wysyła wiadomość w formie 16 znakowego komunikatu do pozostałych serwerów z dodaniem korekcji błędów Reeda-Solomona. Jeżeli wiadomość będzie zbyt uszkodzona i suma kontrolna nie będzie się zgadzała, wówczas klient wysyła żądanie o retransmisję danych. Projekt powinien mieć możliwość ustalenia stopnia uszkodzenia wiadomości na każdej ze ścieżek.

