PROGRAMOWANIE WSPÓŁBIEŻNE I ROZPROSZONE, KOLOKWIUM II

18 stycznia 2017 r. Imię i nazwisko

Tabela odpowiedzi: [WZ6VI2]

Liczba punktów:

PROGRAMOWANIE WSPÓŁBIEŻNE I ROZPROSZONE, KOLOKWIUM II

18 stycznia 2017 r.

т .		
Imıę	1	nazwisko:

Tabela odpowiedzi: [WZ6VI2] Liczba punktów:											w:
1	2	3	4	5	6	7	8	9	10		

- 1. Analiza formalna programów współbieżnych służy do:
 - A. Badania własności systemu związanych z formalnymi wymaganiami użytkownika.
 - B. Implementowania testów jednostkowych systemu.
 - C. Sprawdzania zgodności zachowania modelu ze specyfikacją wejściową.
- W modelu scentralizowanym można stosować zmienne wspódzielone, czyli dostępne dla wielu procesów. Które z poniższych stwierdzeń jest fałszywe:
 - A. Przy braku wsparcia ze strony sprzętu i oprogramowania można wykorzystywać jedynie wysokopoziomowe intstrukcje języka do implementacji synchronizacji.
 - B. Korzystanie z takich zmiennych jest zawsze dozwolone i nie prowadzi do błędów.
 - C. Niekontrolowane modyfikowanie takich zmiennych w wielu procesach jednocześnie moze dawać niedające się przewidzieć efekty.
- 3. Sieci miejsc i przejść zwane Sieciami Petriego definiuje się jako następującą piątkę: (P, T, A, W, s_0) . P i T oznaczją:
 - A. Niepuste znakowanie początkowe (ang. initial marking)
 - B. Rozłączne zbiory miejsc (ang. places) i żetonów (ang. tokens).
 - C. Rozłączne zbiory miejsc (ang. *places*) i tranzycji (ang. *transitions*).
 - D. Funkcje wag i znakowanie początkowe.
- 4. W języku Java klasę wątku można zdefiniować w następujący sposób:
 - A. Definiując klasę implementującą interfejs Runnable zawierającą metodę start().
 - B. Definiując klasę dziedziczącą po klasie Thread bez konieczności implementacji żadnych metod.
 - C. Definiując klasę dziedziczącą po klasie Thread zawierającą metodę start().
 - D. Definiując klasę implementującą interfejs Runnable zawierającą metodę run().
- 5. Znaczniki (ang. tokens) w sieci Petriego:
 - A. Opisują dynamikę sieci.
 - B. Reprezentowane są przez strzałki wychodzące z symbolu miejsca.
 - C. Oznaczają przejście z jednego stanu do drugiego.
- 6. Semafor jest to abstrakcyjny typ danych, na którym można wykonywać dwie operacje:
 - A. W (wait) i N (notify)
 - B. R (wait) i S (signal)
 - C. P(wait) i V(wait)

- A. Rozwiązywać klasycznych problemów współbieżności ponieważ semafory są zbyt prostym mechanizmem.
 - B. Wielokrotnie wykonywać na tym samym semaforze operacji semaforowych, gdyż może to grozić wystąpieniem zakleszczenia.
 - C. Testować wartości zmiennej semaforowej i wykonywać działań arytmetycznych.
- 8. Jeśli tranzycja *t* w Sieci Petriego jest aktywna (suma liczba żetonów w miejscach wchodzących do tranzycji jest większa lub równa sumie wag łuków łączących miejsca z tranzycją *t*), to:
 - A. Tranzycje t na pewno zostanie odpalona.
 - B. Tranzycja t usuwa z miejsc wejściowych tyle znaczników, ile wynosi waga łuku wychodzącego z tranzycji t.
 - C. Odpowiednia liczba znaczników zostaje przetransportowana przez tranzycję *t* do miejsca docelowego.
 - D. Tranzycja t może zostać odpalona.
- 9. Implementując rozwiązanie problemu wzajemnego wykluczania możemy zapisać następujący kod reprezentujący proces.

mutex:

```
binary semaphore := 1;
process P;
begin
  repeat
    własne_sprawy;
    P (mutex);
    sekcja_krytyczna;
    [_____]
until false
```

end

Aby kod procesu był prawidłowy, w pustym miejscu należy wpisać:

- A. V(mutex)
- B. P(mutex)
- C. V(release)
- D. P(release)
- 10. Semafor charakteryzuje się tym, że:
 - A. Każdy proces oczekujący zostanie w skończonym czasie obudzony, chyba że jeden z procesów będzie częściej wykonywał operacje semaforowe.
 - B. Dwie operacje semaforowe to oprócz możliwoście sprawdzania zmiennej semaforowej, jedyne możliwe do wykonania operacje na semaforze.
 - C. Dwie operacje semaforowe to jedyne operacje, które można wykonać na semaforze. W szczególności nie ma możliwości sprawdzenia zmiennej semaforowej.
 - D. Każdy proces oczekujący zostanie zostanie w skończonym czasie obudzony, jeśli tylko semafor zostanie zresetowany dostatecznie dużo razy.

Przy użyciu semafora nie można:

