

Uwagi do egzaminu z Przetwarzania
Rozproszonego, zadania 3 i 4te

5 lipca 2016

Wprowadzenie

Na wstępie chciałbym podziękować wszystkim tym, którzy oddali puste kartki, co znacząco zmniejszyło nakład pracy potrzebny do ich sprawdzenia. Uprzedzając pytanie, moja wdzięczność nie będzie się przekładała na liczbę punktów. Równocześnie czuję się zmuszony do wyrażenia swojego głębokiego niesmaku wobec osób, które zamiast pisać, rysują na kartce bardzo realistyczne wyobrażenie drutu kolczastego (czy ja prosilem o narysowanie wyglądu zasieków, czy co?).

Prace sprawdzane były w ciągu kilku dni, co może wprowadzać niespójności w punktacji. W takim wypadku proszę się koniecznie zgłosić: jeżeli dwie osoby i oraz j otrzymały powiedzmy P_i^x oraz P_j^x punktów za to samo zadanie x , to zaktualizuję punktację w sposób następujący: $P_i^x = \min(P_i^x, P_j^x)$.

Tradycyjnie już, za pisanie głupot były odejmowane punkty cząstkowe. Uzasadnienie: zadaniem egzaminu jest sprawdzenie waszej wiedzy. Jeżeli ktoś z własnej, nieprzymuszonej woli objawia swoją niewiedzę, ułatwia to nasze zadanie.

Zadanie trzecie

Ocena zadania trzeciego jest najbardziej subiektywna. Zadanie to jest w gruncie rzeczy bardzo proste, ponieważ poprawność stwierdzenia jest w intuicyjny sposób oczywista dla każdego znającego algorytm implementujący zegary wektorowe Matterna. Celem zadania było więc sprawdzenie, czy student potrafi w sposób przejrzysty i formalny zapisać rozumowanie (im bardziej formalnie i szczegółowo, tym lepiej). Sam fakt, że prawdziwość twierdzenia jest oczywista, powinna być wskazówką, że należy także oczywiste rzeczy pisać (np. punkt (a) poniżej - oczywistość, ale trzeba ją zapisać, żeby pokazać, że się o tym wie). Każdy brak precyzji był karany odejmowaniem punktów.

Zasadniczo można było rozważyć dwa przypadki (plus ewentualnie trzeci przypadek dla zdarzeń z tego samego procesu) i dla każdego z nich wykazać, że muszą się one różnić na conajmniej jednej pozycji (dla zdarzeń niezależnych, oraz dla zdarzeń zależnych). Na przykład

(a) dla zdarzeń należących do tego samego procesu P_i , algorytm wymusza, że przy każdym zdarzeniu (powiedzmy, E_i) zawsze zwiększamy wektor na i -tej pozycji.

(b) z algorytmu wynika, że kiedy proces P_i zinkrementuje pozycję i -tą w swoim zegarze wektorowej w wyniku zdarzenia E_i , procesy $P_{k \neq i}$ mogą się o tym "dowiedzieć" tylko po otrzymaniu wiadomości z odpowiednią etykietą wektorową. Tak więc, w momencie aktualizacji wartości i -tej przez P_i na wartość m , wartość i -ta zegara wektorowego na każdym innym procesie $P_{k \neq i}$ nie może być równa m (dokładniej, musi być mniejsza od m), i wartość ta może być większa lub równa m tylko dla zdarzeń zależnych od E_i .

(c) z algorytmu wynika, że w efekcie odbioru wiadomości, w której wartość i -ta wynosi m , proces P_k zinkrementuje wartość k -tą w etykietce wektorowej (powiedzmy, na x) odpowiadającej zdarzeniu odbioru E' , gdzie oczywiście $E_i \mapsto E'$. Z (a) wynika, że pozycja k -ta etykiety wektorowej odpowiadającej E_i musi być oczywiście mniejsza niż x , a więc etykiety odpowiadające E_i oraz E' muszą być różne, z (b) wynika, że w chwili zajścia E' , k -ta pozycja zegarów wektorowych wszystkich innych procesów musi być mniejsza (a więc różna) od x .

Z algorytmu wynika, że pozycje w wektorowych etykietach zegarowych są niemalejące. Z (c) wynika, że zdarzenia poprzedzające przyczynowo E_i muszą mieć wartość i -tej pozycji etykiety wektorowej mniejsze niż m (a więc różne). Również dla zdarzeń niezależnych z E_i , z (b) wynika, że wartość i -tej pozycji ich etykiet wektorowych musi

być mniejsza niż x . Dla zdarzeń zależnych od E_i , z (c) rekurencyjnie możemy wywieść, że ich etykiety muszą być różne od etykiety E_i .

Poprawne rozumowanie musi zawierać wszystkie elementy powyższego rozumowania: a więc rozważać nie tylko dwa zdarzenia (np wysłania i odbioru), nie pomijać zdarzeń jednego procesu, nie pomijać zdarzeń niezależnych. Istnieje możliwość, że przez pomyłkę wstawiłem komuś trzy punkty, mimo, że nie powinienem (bo zapomniał/a np. o zdarzeniach niezależnych). Można to zgłosić i oczywiście natychmiast wtedy swój błąd poprawię obniżając punktację.

Sam opis algorytmu, zakończony “z tego wynika X” nie wystarcza, ponieważ brak zapisu rozumowania prowadzącego z opisu algorytmu do wniosku. Nie wystarcza wspomnieć, że “przy każdym odbiorze inkrementujemy na i -tej pozycji, więc zawsze będą wektory różne”, bo należało dodać, że proces $P_{k \neq i}$ nie może sam z siebie inkrementować i -tej pozycji. Tak samo, jeżeli ktoś wykaże, że dla jednego przypadku (na przykład wysłania i odebrania wiadomości) zegary na pewno są różne, to za mało: należy uzasadnić, że analiza tylko tego przypadku wystarcza, albo wymienić i zaanalizować pozostałe przypadki. Dodatkowo, jeżeli ktoś napisał, że “u procesu i -tego zawsze wartość i -ta wektora jest większa niż u innych procesów”, miał odejmowane punkty (bo to nieprawda). Tak samo, jeżeli pisał/a “ponieważ nadawca zwiększa i -tą wartość wektora, więc jego etykieta zegarowa jest różna od odbiorcy”, albo “zdarzenia modyfikują tylko swoje pozycje w wektorze” (zdarzenia?!?). Punkty były też odejmowane za stwierdzenie “wartość zegara jest zwiększana tylko przy zdarzeniach odbioru/wysłania” (a zdarzenia wewnętrzne to pies?).

Z uwagi na dużą subiektywność oceny w tym zadaniu oraz fakt, że punkty odejmowałem za brak wystarczająco formalnie zapisanego rozwiązania (bo “czwórkowicze” oraz — zwłaszcza — “piątkowicze” powinni pisać klarownie i formalnie), jeżeli komuś będzie brakować do 1,5 punkta do zaliczenia, jestem otwarty na dyskusję (jeżeli ktoś miał 1,5 z tego zadania, a ja jemu/jej odebrałem punkty z uwagi na braki wspomniane powyżej). Zastrzegam jednak, że mogę nie zwiększyć punktacji, albo mogę ją powiększyć o liczbę mniejszą niż wymagana do zaliczenia, oferta dotyczy się tylko punktów brakujących do zaliczenia (a nie np. kiedy komuś brakuje pół punkta do piątki).

Zadanie czwarte

W zadaniu czwartym należało wyliczyć i uzasadnić oba przypadki. Odpowiedź typu “dla pesymistycznej złożoność komunikacyjna pakietowa to n^2 a złożoność czasowa to n ”) była punktowana zerem (bo nie widać wyliczenia ani uzasadnienia odpowiedzi, ani wyjaśnienia jaki to przypadek optymistyczny, a jaki pesymistyczny. Za samo poprawne wyjaśnienie i wyliczenie jednego przypadku dawane było 2 punkty. Nie były przyznawane żadne punkty za odpowiedzi niezwiązane z tematem. Czasami zdarzały się poprawne wyniki, ale nijak się mające do wyliczeń, np. ktoś pomylił algorytm pasywny i aktywny, po czym na dodatek napisał “proces wysyła do wszystkich, potem wszystkie do pozostałych, a więc złożoność czasowa 1” — owszem, jeden, ale dlatego bo wszystkie dostarczają już po otrzymaniu pierwszej wiadomości. Wyjątek: jedna osoba napisała, że będzie analizować algorytm aktywny, osoba ta otrzymała 2.75 punkta.

Ewidentnie część z państwa nie uważało na wykładzie; zauważyłem, że część sądzi, jakoby wykrywany był błąd ostatniego nadawcy wiadomości, a przecież chodzi tylko o oryginalnego nadawcę i wykrycie jego błędu. Przypomnijmy algorytm: Inicjator P_i wysyła pakiet $pckt$, $pckt.origin = P_i$. Załóżmy, że dostał ten pakiet tylko P_j oraz P_k , po czym P_i uległ awarii. Po wykryciu tego przez P_j , rozsyła on wiadomość i powiedzmy, że ulega awarii. P_k wykrywa awarię P_i i rozsyła wiadomość, a następnie otrzymuje wiadomość od P_j oraz następnie wykrywa awarię P_j . Wówczas, ponieważ $pckt.origin \neq P_j$,

P_k nie rozsyła drugi raz pakietu. Powiedzmy teraz, że jakiś proces wykrył awarię P_i zanim otrzymał wiadomość, po czym otrzymał pakiet w którym $pckt.origin = P_i$. Oczywiście wówczas natychmiast, zgodnie z linijką 12 (slajd 19 w mechanizmy przetwarzania rozproszonego) wysyła wiadomość. Powiedzmy, że jakiś proces najpierw otrzymał wiadomość, w której $pckt.origin = P_i$, po czym długo, długo później wykrywa dopiero awarię P_i . Wtedy zgodnie z linijką 17 w następnym slajdzie rozsyła wiadomość. Powiedzmy, że proces P_x otrzymał wiadomość, dostarczył ją, ale zanim wykrył awarię P_i sam uległ awarii. Nic się nie stało, ponieważ własności zgodnego dostarczania mówią tylko, że jeżeli poprawny proces dostarczył wiadomość, to wszystkie inne poprawne procesy mają dostarczyć wiadomość (a skoro P_x uległ awarii, to nie był poprawny).

Najgorszy przypadek, to gdy procesy ulegają po kolei awarii, wysyłają każdy po n wiadomości, ale tylko jedna z nich dochodzi do jednego z procesów, który następnie wysyła n wiadomości, ulega awarii itd, aż zostaje jeden poprawny proces. Bardzo dużo osób zakończyło na tym analizę i nie napisało nic o przypadku optymistycznym, albo nie uzasadniło złożoności dla przypadku optymistycznego. W przypadku tym drugim dawałem 2 punkty — gdy komuś brakuje nie więcej niż 1 punkta do zaliczenia, jestem skłonny te brakujące 1 punkta dodać (uwaga jak wyżej: musi brakować DOKŁADNIE punkt do zaliczenia). Oferta dotyczy tylko jeżeli ktoś poprawnie wyliczył złożoność do przypadku najgorszego, i podał złożoność nie uzasadniając jej dla przypadku optymistycznego (co można poznać po tym, że ma dwa punkty z tego zadania). Jeżeli ktoś wyliczył poprawnie złożoność dla przypadku najgorszego, ale zapomniał o przypadku najlepszym, dostał 1,5 punkta - jeżeli w tym wypadku komuś brakuje dokładnie pół punkta do trójki, jestem skłonny te pół punkta dodać.

UWAGA: to ja decyduję, czy student kwalifikuje się do przypadków wymienionych wyżej. Proszę nie żebrać “brakuje mi tylko punkt, a tamtym obiecał Pan dodać”. W tych dwóch przypadkach uznaję, że ktoś potrafił poprawnie wyliczyć złożoności i zidentyfikować przypadki najlepszy/najgorszy, a więc nie ma co z powodu drobnej pomyłki “uważać” studenta. Jeżeli komuś brakuje ćwiartki punkta, ale nie kwalifikuje się do wypadków wymienionych powyżej, prędzej utnę jakiegoś punkta niż dodam, jeżeli ktoś przyjdzie, mówiąc brzydko, “zebrać”. Przy czym “zebranie” jest wtedy, kiedy ktoś przychodzi i mówi “brakuje mi tylko pół punkta, niech pan mi doda...”. Jeżeli ktoś przychodzi i mówi “moim zdaniem tutaj się pan pomylił i powinienem dostać dodatkowe pół punkta, oto moje argumenty” to nie jest, oczywiście, zebranie.

Za niektóre błędy, jak zwykle, dawałem minusy. Niektóre ciekawe błędy:

- “detektor wykrywa nieistniejącą awarię” — algorytm pasywny wymaga doskonałego detektora awarii, który się nie myli.
- “detektor wykrywa, że wysłał wiadomość do niepoprawnego procesu” — bzdura. Wysłanie wiadomości do niepoprawnego procesu nie ma żadnego znaczenia.
- “najgorszy przypadek to taki, w którym wszystkie procesy ulegają awarii” — nie analizujemy takich przypadków.
- “inicjator wysyła wiadomości do wszystkich, każdy z n procesów wysyła do wszystkich” — to miał być algorytm pasywny, nie aktywny. UWAGA: jeżeli ktoś pomylił algorytm pasywny z aktywnym, ale poprawnie powycliczał złożoności, otrzymał 1,5 do 2 punktów. W takim wypadku, jeżeli komuś brakuje punktów do trójki, punkty zostaną z automatu przyznane pod warunkiem, że (a) brakuje nie więcej niż 1.5 punkta do trójki, tj. jeżeli komuś brakuje 2 punktów, nie ma co przychodzić (b) wyliczenie złożoności musi być poprawne.
- “jeżeli procesy ulegną awarii, to ich monitory będą propagować wiadomości...” — kapitalne. Zakładamy, że monitor ulega awarii razem z procesem, więc tutaj mamy ewidentny przypadek życia pozagrobowego procesów.

- “ostatni proces musi wysłać n wiadomości do n procesów” — kolejna perełka. Dlaczego niby JEDEN proces ma wysłać n wiadomości do każdego z pozostałych procesów???
- “po awarii procesu kolejny bierze na siebie rozgłaszanie” — jaki kolejny? Skąd wiadomo, który?

Najwyższa wyliczona złożoność czasowa w tym roku to n^2 . Ho, ho...