

# Symulacja cyfrowa - projekt zaliczeniowy

## Zadanie 1b1, metoda przeglądania działań

Gliwiński Jarosław Marek  
Grupa dziekańska I5  
#74839

20 czerwca 2008

### 1 Treść zadania

Napisać program symulacyjny badający właściwości systemu kontroli jakości w firmie produkcyjnej. Firma wytwarza 3 rodzaje produktów. Każdy z produktów przechodzi przez dwie fazy kontroli jakości. Istnieje łącznie 10 stanowisk kontroli jakości. W pierwszej fazie kontroli produkt może być sprawdzany na dowolnym stanowisku kontrolnym. W drugiej fazie kontroli, produkt musi być sprawdzany na stanowisku wyspecjalizowanym. Istnieje łącznie po trzy wyspecjalizowane stanowiska do badań w fazie drugiej dla każdego produktu. Wykrycie wady w pierwszym etapie powoduje skierowanie produktu do poprawki, po czym ponownie trafia on do kontroli etapu pierwszego. Wada na drugim etapie powoduje odrzucenie produktu. Firma wytwarza jednakową ilość każdego produktu. Produkty są dostarczane do kontroli indywidualnie a czas między kolejnymi zgłoszeniami produktu jest zmienną losową o rozkładzie wykładniczym. Czas produkcji w fazie drugiej jest zmienną losową o rozkładzie Gaussa o wartości oczekiwanej 85 min i wariancji 1 min.

#### Założenia

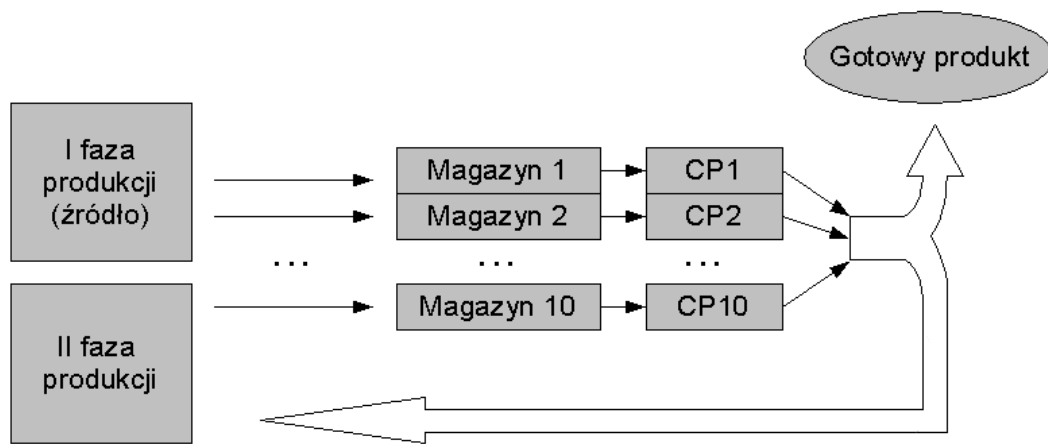
1. Czas kontroli w fazie pierwszej jest zmienna losową o rozkładzie jednostajnym w przedziale od 5 do 15 minut.
2. Czas kontroli w fazie drugiej jest zmienną losową o rozkładzie wykładniczym i wartości oczekiwanej 10 min.
3. Czas wykonania poprawki jest zmienna losową o rozkładzie Gaussa o wartości oczekiwanej 10min i wariancji 5min.
4. Produkty badane w fazie pierwszej mają pierwszeństwo podczas kontroli.
5. Każde stanowisko kontroli posiada magazyn na 20 produktów.
6. Brak miejsca w magazynach stanowisk kontroli wstrzymuje całą produkcję.
7. Prawdopodobieństwo braków na pierwszym etapie to 1%, a na drugim 0.5%.

**Zadanie** Za pomocą symulacji określić:

1. Maksymalna wydajność fabryki (produkty/godzinę)
2. Średnią zajętość magazynów w każdym punkcie kontrolnym.
3. Średnie wykorzystanie każdego punktu kontrolnego.

Zastosowano metodę przeglądania działań.

**Schemat blokowy systemu**



## 2 Struktura programu

### 2.1 Implementacja modelu

Podstawowe elementy systemu takie jak źródło produktów, stanowiska kontroli zaimplementowano jako obiekty. Podstawowe zaimplementowane klasy obejmują:

**checkpoint** zawiera podstawowe zmienne charakteryzujące punkt kontrolny (rodzaj specjalizacji, czas zakończenia kontroli), udostępnia magazyn i dokonuje przekazywania dalej przetwarzanych produktów. Każde ze stanowisk posiada własny generator pseudolosowy.

**product** to klasa reprezentująca pojedynczy produkt. Przede wszystkim przechowuje informacje o typie i statusie (przeszedł/nie przeszedł I fazy kontroli) produktu. Warto nadmienić, że zdefiniowano operator mniejszości porównujący produkty, dzięki czemu można prawidłowo obsługiwać kolejkę priorytetową z STL.

`source` reprezentuje pierwszy etap produkcji. Obiekt tej klasy generuje wskaźniki do obiektów `product`, które są przekazywane pomiędzy obiektami pozostałych klas. Wartość oczekiwana czasu produkcji jest pobierana jako parametr konstruktora.

`generator` to oczywiście klasa udostępniająca szereg generatorów pseudolosowych

## 2.2 Zdarzenia w systemie

Można wyróżnić następujące zdarzenia czasowe:

1. Pojawienie się w systemie nowego produktu ze źródła
2. Zakończenie kontroli i zwolnienie punktu kontrolnego
3. zakończenie drugiej fazy produkcji

Istotnym pośrednim skutkiem zdarzeń 1 bądź 3 może być blokada produkcji wynikająca z braku miejsca w magazynach punktów kontrolnych.

Z kolei zdarzenie warunkowe występuje jedno: pobranie produktu z kolejki do wolnego stanowiska kontroli.

Dodatkowo obsługiwane jest niezwiązane bezpośrednio z modelem symulacyjnym zdarzenie związane z próbkowaniem statystyk.

## 2.3 Obsługa z linii poleceń

Program może działać, w zależności od parametrów, w następujących trybach, umożliwiających powtórzenie wyników zawartych w raporcie:

`seeds plik_wynikowy` generuje do pliku wskazanego parametrem 1000 kolejnych jąder generatora pseudolosowego, co 250000 kroków.

`max_eff plik_wynikowy ilosc_produktow(warunek_konca)` pozwala generować wyniki pozwalające określić maksymalną efektywność systemu w badanym przedziale

`mean_store plik_wynikowy ilosc_produktow(warunek_konca) wart.oczekiwana seed` pozwala generować wyniki w postaci średniej zajętości magazynów

`mean_usage plik_wynikowy ilosc_produktow(warunek_konca) wart.oczekiwana seed` pozwala generować średniego użycia stanowisk kontroli w skali czasu

Generalnie wiele wartości niezmiennych w toku eksperymentu -takich jak rozkłady czasów kontroli czy ilość stanowisk jest wpisanych „na sztywno” w kodzie programu.

## 2.4 Generatory pseudolosowe

Implementacje generatorów oparte są na informacjach z książki „Object-oriented Computer Simulation of Discrete-Event Systems” autorstwa prof. Jerzego Tyszera.

Użyto następujących generatorów:

`UniformPrototype()` podstawowy generator wytwarzający liczby o rozkładzie równomiernym z zakresu liczb całkowitych reprezentacji 32bit

`Uniform01()` rzutowanie powyższego na przedział 0-1

`UniformFromTo(double from, double to)` jak wyżej, przedział `from-to`

`Exponential(double rate)` rozkład wykładniczy, warto nadmienić, że pobieranym przez funkcję parametrem jest wartość oczekiwana

`Bernoulli2(double failureChance)` rozkład dwupunktowy Bernoulliego

`GaussPrototype(double mu, double sigma)` rozkład normalny

`Gauss(double mu, double sigma)` jak wyżej, dodatkowo zabezpieczony przed generacją liczb ujemnych (na potrzeby losowania czasu)

W program wbudowano tablicę 1000 jąder, wyborem których można sterować z wiersza poleceń. Każde stanowisko kontrolne i źródło produktów posiadają własne obiekty klasy `generator`. Podawane z linii poleceń jądro jest jądrem początkowym, w toku działania programu kolejne przebiegi symulacji korzystają z kolejnych wartości z tablicy – tj. podawana z linii poleceń wartość jest pierwszą z listy użytych.

Drobną modyfikacją w kodzie programu można uruchomić funkcje testujące generatory. Pomimo zastosowania powszechnie uznanych generatorów z literatury, dla sprawdzenia poprawności implementacji wykonano testy, które wypadły pomyślnie.

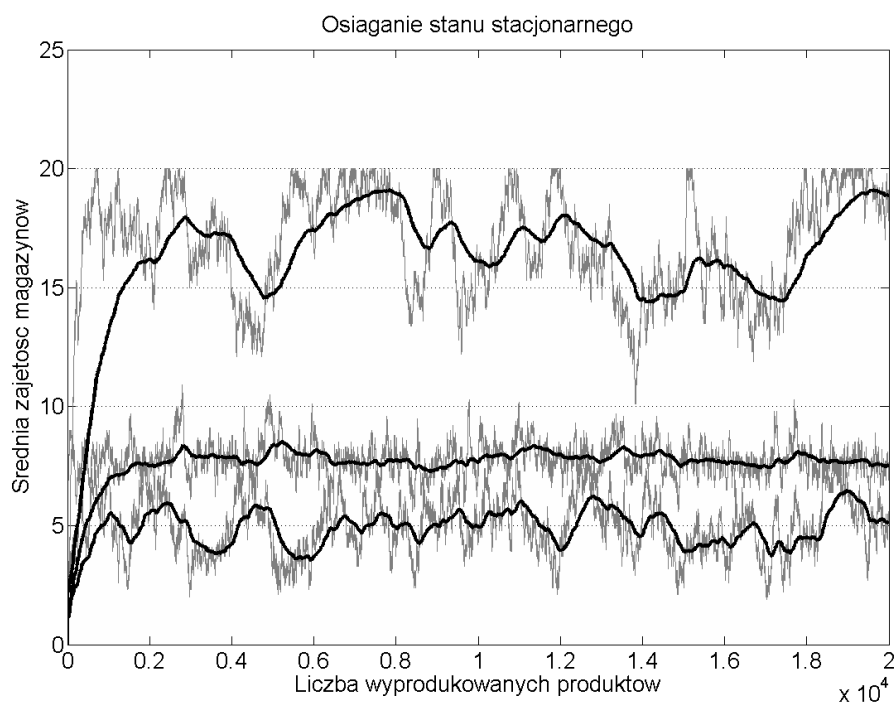
## 3 Symulacja

Założenia:

- poziom istotności:  $\alpha = 0.95$
- liczba przebiegów symulacji:  $n = 10$
- przedział ufności zawierający się w przedziale średnia  $\pm 5\%$

### 3.1 Okres przejściowy

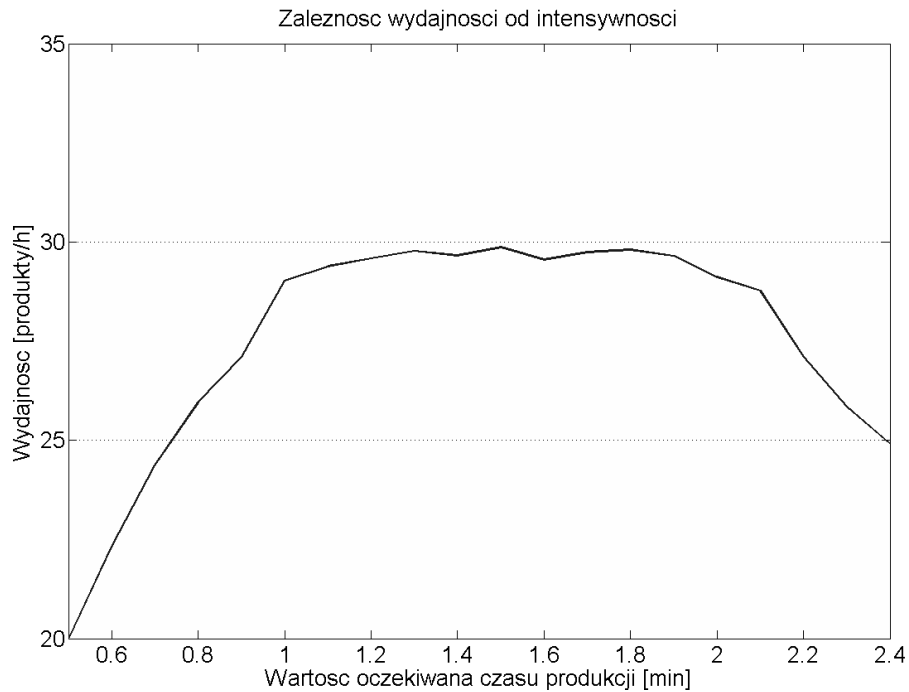
Pierwszym krokiem było stwierdzenie po jakim okresie działania system znajdzie się w stanie, który będziemy mogli określić jako stacjonarny. Po przeprowadzeniu prób dla kilku intensywności stwierdzono, z dużą dozą pewności można określić, że powyżej 2000 wyprodukowanych produktów system znajduje się w stanie stacjonarnym. Poniżej znajduje się wykres potwierdzający tę zależność, wykreślony przykładowo dla źródeł produktów o wartościach oczekiwanych czasu produkcji odpowiednio 2, 3, 4 minuty:



### 3.2 Maksymalna wydajność

We wstępnej bazie badania założono jako warunek końca 20 tysięcy wyprodukowanych poprawnych produktów.

W celu określenia maksymalnej wydajności przeprowadzono 20 prób z zakresu wartości oczekiwanych  $\langle 0.5; 2.4 \rangle$  z krokiem 0.2. Uzyskane wartości ilustruje następujący wykres:



Jak widać, pomijając fluktuacje związane z losowością poszczególnych pomiarów, środek przedziału maksymalnej wydajności przypada dla wartości oczekiwanej około 1.5. Tę właśnie wartość przyjęto do następnych badań.

### 3.3 Średnia zajętość magazynów

Tabela przedstawia wyniki zajętości magazynów stanowisk kontroli dla wartości oczekiwanej 1.5:

<i>lp</i>	Średnia zajętość magazynów										średnia
1	19.8821	19.8310	19.8106	19.8312	19.7132	19.7078	19.7414	19.5754	19.5539	19.6148	19.7261
2	19.8866	19.8235	19.8163	19.8035	19.7602	19.7190	19.7191	19.6312	19.6026	19.5873	19.7349
3	19.8858	19.8152	19.8102	19.8133	19.6956	19.7304	19.7072	19.5712	19.5935	19.6018	19.7224
4	19.8840	19.8153	19.7989	19.8151	19.7061	19.6942	19.6905	19.5826	19.5648	19.5090	19.7060
5	19.8824	19.8172	19.7999	19.8160	19.7356	19.6927	19.7118	19.5731	19.5023	19.5242	19.7055
6	19.8858	19.8386	19.8334	19.8402	19.7698	19.7494	19.7648	19.6220	19.6526	19.6314	19.7588
7	19.8889	19.8045	19.8091	19.8337	19.6925	19.7115	19.7585	19.5671	19.5888	19.6440	19.7299
8	19.8884	19.8309	19.8285	19.8269	19.7276	19.7366	19.7104	19.5635	19.5792	19.5317	19.7224
9	19.8876	19.8270	19.8321	19.8244	19.7474	19.7445	19.7467	19.6745	19.6690	19.6492	19.7602
10	19.8811	19.8268	19.8205	19.8355	19.7290	19.7366	19.7625	19.6229	19.5667	19.6523	19.7434

Skrajna prawa kolumna reprezentuje uśrednienie po wszystkich magazynach w ramach jednej symulacji (symulacje numerowane kolumną *lp*).

Przedział ufności wyniósł  $\langle 19.7174, 19.7445 \rangle$ , co przy średniej  $19.7310 \pm 5\%$  zamykającej się w zakresie  $\langle 18.7444, 20.7175 \rangle$  jest wynikiem w zupełności wystarczającym.

Jak widać pierwsza założona wartość (20000) definiująca koniec symulacji prowadzi do przedziału ufności mieszczącego się w żądanym zakresie. Dzięki temu wyniki uzyskane w tym i poprzednim punkcie możemy od razu uznać za poprawne w skali eksperymentu. Dla tak zdefiniowanych warunków początku i końca pozostało jeszcze przeprowadzenie ostatniej analizy.

### 3.4 Średnie wykorzystanie punktów kontrolnych

Ostatnie badanie polegało na określeniu dla wyznaczonej wcześniej intensywności wykorzystania punktów kontrolnych. Poniżej tabela reprezentująca te wyniki:

<i>lp</i>	Średnie wykorzystanie stanowisk										średnia
1	0.999882	0.999937	0.999829	1.000000	0.998659	0.999724	1.000000	0.999394	0.999791	0.999656	0.9997
2	0.999872	0.99999	0.999935	0.999486	0.999832	0.999933	0.999559	0.999838	0.999781	1.000000	0.9998
3	1.000000	0.999987	0.999924	0.999931	0.999698	0.99988	1.000000	0.999917	0.9999	0.999875	0.9999
4	0.999901	0.999847	0.999892	0.999749	0.999978	0.999994	0.999775	0.999388	0.999913	1.000000	0.9998
5	0.999887	0.999813	0.999803	0.999953	0.999917	0.999989	0.999646	0.999876	0.999795	1.000000	0.9999
6	0.999789	0.999842	0.999885	0.999636	0.999768	0.999854	0.999803	0.999868	1.000000	0.99972	0.9998
7	0.999921	0.99996	1.000000	0.999784	0.999932	0.99985	0.999836	0.999849	0.999802	0.999096	0.9998
8	0.999875	0.99983	0.999581	0.999538	0.999928	0.999908	0.999455	0.999981	1.000000	0.999965	0.9998
9	0.999967	0.99966	1.000000	0.999525	0.999845	0.999134	0.999539	0.999705	0.999933	0.999302	0.9997
10	0.999707	0.999999	0.999641	0.999844	0.999922	0.999989	0.999038	0.999805	0.99972	1.000000	0.9998

Ponownie skrajna prawa kolumna reprezentuje uśrednienie po wszystkich stanowiskach w ramach jednej symulacji (symulacje numerowane kolumną *lp*).

Jak widać przedział ufności jest bardzo wąski (około  $\langle 0.9997; 0.9999 \rangle$ ), a to dzięki górnemu ograniczeniu wykorzystania stanowisk, które przy maksymalnej wydajności systemu jest niemalże osiągnięte empirycznie.

## 4 Wnioski

Jak pokazują wyniki symulacji system będzie działał optymalnie dla pewnego zakresu intensywności produkcji. W Przypadkach spoza tego zakresu mamy do czynienia z jednym z dwóch przypadków:

- intensywność produkcji jest zbyt mała i nie wykorzystuje w pełni możliwości systemu
- intensywność produkcji jest zbyt duża i brak miejsca w magazynach blokuje produkcję

W zakresie pracy optymalnej system maksymalnie wykorzystuje swoje zasoby - stanowiska kontrolne pracują właściwie bez przerwy, magazyny są całkowicie wypełnione.

Nieprecyzyjnie modelowana jest druga faza produkcji - jedyny parametr to czas, podczas gdy w rzeczywistości rozsądnym mogłoby być uwzględnienie pojemności magazynu, bądź maksimum wydajności równoległego przetwarzania w tej fazie.

Ponadto należy odnotować, iż w praktyce narzędzie symulacji cyfrowej teoretycznie umożliwia symulację niemalże dowolnie złożonych układów, jednakże należy mieć na uwadze ograniczenia wynikające zarówno z kosztu w złożoności obliczeniowej implementacji nadmiernej liczby szczegółowych parametrów, jak i sensowność stosowania ich, podczas gdy ich wpływ na wyniki eksperymentu symulacyjnego może okazać się marginalny