

# Hurt bitów

## "Producent & Son Co."

(2020/2021)

## 1. Wstęp

Należy przygotować dwa programy, producenta i konsumenta, współpracujące ze sobą w układzie: jeden producent wielu konsumentów. Producent musi być gotowy na równoczesną obsługę nawet setek klientów.

Producent „produkuje bajty” i umieszcza je w „magazynie”. Zawartość magazynu jest udostępniania konsumentom za pomocą protokołu TCP/IP.

## 2. Producent

```
program:          producent
parametry:
    -p <float>      tempo wytwarzania materiału podawane w jednost-
                    kach  $\frac{2662B}{sec}$ ,
    [<addr>:]port    lokalizacja producenta
                    domyślną wartością dla <addr> jest łańcuch
                    "localhost"
```

### 2.1. Produkcja

Produkcja odbywa się blokami, z których każdy jest rozmiaru 640 bajtów. Każdy blok zawiera bajty jednakowej wartości, a wartości stosowane w kolejnych blokach przebiegają cyklicznie przez wszystkie małe i wielkie litery ASCII. Czas potrzebny na wyprodukowanie bloku jest stały i wynika z wartości parametru -p (wielkość bloku  $\times$  tempo produkcji).

Jeżeli produkcja wypełni cały magazyn, to zostaje wstrzymana do czasu pojawienia się wolnego miejsca, pozwalającego na umieszczenie całego bloku.

### 2.2. Dystrybucja

Równocześnie z produkcją obsługiwane są połączenia TCP/IP, za pomocą których zawartość magazynu jest wydawana klientom. Po akceptacji połączenia, producent używa go do przesłania z magazynu porcji produktu o wielkości 13 KiB. Produkt jest wysyłany kawałkami, których wielkość nie przekracza 4 KiB. Warunkiem rozpoczęcia transmisji jest posiadanie w magazynie odpowiedniej ilości zasobów (13 KiB). Jeżeli ich brak, producent wstrzymuje rozpoczęcie realizacji zamówienia, do momentu spełnienia tego warunku. (Liczba bajtów, potrzebnych do realizacji już

rozpoczętych transmisji, jest odliczana od zajętości magazynu.)

Po zakończeniu transmisji, producent zamyka połączenie.

Jeżeli przed rozpoczęciem transmisji klient zerwie połączenie, to zamówienie jest anulowane. Natomiast, jeżeli zerwanie nastąpi już w jej trakcie, to i tak odpowiednia ilość bajtów zostanie z magazynu pobrana i „zmarnowana”.

### 2.3. Realizacja magazynu

Magazyn producenta ma być zrealizowany za pomocą potoku (pipe). Aby zapewnić poprawne działanie mechanizmu synchronizacji, producent korzysta z pomocy potomka. Jeden z tych procesów odpowiada za produkcję, a drugi za dystrybucję produktów między klientów. Łączący je potok (magazyn) gwarantuje blokowanie i atomowość operacji.

## 3. Klient

```
program:          konsument
parametry:
  -c <int>         pojemność magazynu konsumenta liczona w blokach
                   o rozmiarze 30 KiB,
  -p <float>        tempo konsumpcji danych, podawane w jednostkach
                    $\frac{4435\text{B}}{\text{sec}}$ ,
  -d <float>        tempo degradacji materiału, podawane w jednostkach
                    $\frac{819\text{B}}{\text{sec}}$ ,
  [<addr>:]port     lokalizacja producenta
                   domyślną wartością dla <addr> jest łańcuch
                   "localhost"
```

Aby otrzymać produkty, konsument nawiązuje połączenie z producentem. Następnie odbiera dostarczane dane. Każda porcja bajtów jest „przetwarzana” w tempie określonym parametrem -p, co oznacza, że konsument czeka zanim pobierze kolejną porcję.

Przetworzone dane zwiększają zajętość lokalnego magazynu. Jednocześnie, materiał umieszczony w magazynie ulega degradacji (biodegradacja bajtów), w tempie określonym parametrem -d. Oznacza to, że w czasie oczekiwania na nowe dane od producenta, dotychczasowa zajętość magazynu będzie malała.

Po zakończeniu transmisji, konsument sprawdza czy zostało mu dość wolnego miejsca (13 KiB), i na tej podstawie decyduje czy pobrać kolejny blok, czy zakończyć działanie. Ze względu na zjawisko degradacji, nie można z góry założyć, ile bloków danych konsument będzie w stanie wczytać.

**Uwaga.** Lokalny magazyn nie musi być fizycznie implementowany. Wystarczy operować na jego pojemności i zajętości.

## 4. Raporty

Oba programy wypisują raporty na standardowy strumień diagnostyczny. W przypadku producenta są one generowane w trzech momentach:

1. po każdym rozłączeniu: *TS*, adres rozłączonego klienta, ilość zmarnowanych bajtów,
2. co 5 sek.: *TS*, ilość podłączonych klientów, zajętość magazynu (nominalnie i procentowo), przepływ materiału w czasie ostatnich 5-ciu sekund (ilość materiału wygenerowanego minus ilość wydanego).

Natomiast konsument generuje raport, gdy rezygnuje z dalszego pobierania produktów i kończy działanie. Zawiera on:

1. *TS*,
2. identyfikację konsumenta: PID oraz adres, z którego się łączył,
3. dla każdego bloku z osobna:
  - a) opóźnienie między połączeniem, a odebraniem pierwszej porcji produktu,
  - b) czas między pierwszą porcją a zamknięciem połączenia.

Stosowane powyżej *TS* oznacza odczyt z zegara `CLOCK_REALTIME`. Natomiast opóźnień są wyznaczane na podstawie zegara `CLOCK_MONOTONIC`.