

Piotr Chrzastowski-Wachtel  
Uniwersytet Warszawski

# *Dziedziny algorytmiczne*

# Ważne sprawy, które poruszyliśmy

## ■ Dziedzina algorytmiczna

- Aby mówić o algorytmach, należy zawsze mieć na uwadze dokładny repertuar podstawowych środków.
- W przypadku konstrukcji geometrycznych obiektami były punkty, proste, okręgi, wraz z pięcioma dopuszczalnymi operacjami opisanymi na poprzednich slajdach

# Dziedziny algorytmiczne dla algorytmu Euklidesa

- W obu algorytmach Euklidesa używaliśmy nieco innych operacji:
  - W algorytmie Euklides 1:  $(\mathbb{N}, 0, >, -, \text{zamień})$
  - W algorytmie Euklides 2:  $(\mathbb{N}, >_0, \mathbf{mod})$
  - W algorytmie Euklides 3:  $(\mathbb{N}, 0, >, \text{Par}, *2, /2, -)$

# Dziedzina algorytmiczna

- Podstawowym pojęciem algorytmiki jest dziedzina algorytmiczna, czyli system relacyjny

$$(A, \{\sigma_i\}_{i \in I}, \{\tau_j\}_{j \in J})$$

- gdzie zbiór  $A$  jest nośnikiem, zaś zbiory

$$\{\sigma_i\}_{i \in I}, \{\tau_j\}_{j \in J}$$

- są odpowiednio zbiorami operacji i relacji na nośniku.

# Programowanie strukturalne

- Przy programowaniu strukturalnym dziedziny tworzymy hierarchicznie, pozwalając dziedzinie wyższego poziomu korzystać z operacji poziomu niższego.
- Chodzi o to, żeby na żadnym poziomie nie bawić się zbytnimi szczegółami
- Ten styl programowania jest charakterystyczny dla języków nowszych generacji (obiektowe, funkcyjne, w logice).

# Złożoność obliczeniowa

- To było drugie, niezwykle ważne pojęcie algorytmiki.
- Chodzi o liczbę operacji, czyli koszt obliczeń.
- Można ją rozważać w stosunku do
  - algorytmu
  - problemu algorytmicznego.

# Co zliczamy badając złożoność?

- Trudno byłoby skupiać się na każdej operacji wykonywanej w czasie działania algorytmu.
- Wybiera się zatem taką, która wykonuje się najczęściej, a przy tym jest najdroższa (np. operacja mod w algorytmie Euklides 2) i liczy się liczbę jej wystąpień dla pewnych danych..

# Złożoność pesymistyczna a średnia

- Najczęściej jako dane przyjmuje się jeden z 2 wariantów:
  - dane złośliwe (złożoność pesymistyczna)
  - dane typowe (złożoność średnia)



# Złożoność pamięciowa

- Czasem do wykonania algorytmu potrzeba nam dodatkowej pamięci. Rozważamy wtedy złożoność pamięciową i podobnie jak poprzednio dzielimy ją, w zależności od danych, na:
  - złożoność pamięciową pesymistyczną
  - złożoność pamięciową średnią

# Złożoność problemu

- W końcu mówimy też o złożoności problemu; jest to złożoność najlepszego algorytmu w danej klasie rozwiązującego ten problem. Tu też wyróżniamy:
  - złożoność problemu pesymistyczną
  - złożoność problemu średnią

# Złożoność obliczeniowa

- Zatem mamy trzy kategorie złożoności, które można rozważać niezależnie:

czas – pamięć

pesymistyczna – średnia

algorytm – problem

# Rozmiar danych

- Rozmiar danych liczymy w odniesieniu do liczby bitów koniecznych do reprezentowania tych danych
  - dla tablic – ich długość
  - dla liczb – liczba cyfr
  - dla grafów – łączna liczba węzłów i krawędzi

# Podsumowanie

- Algorytmika jest sercem informatyki
- Nie można programować byle jak. Nawet najszybsze komputery nie poradzą sobie z programami napisanymi niechlujnie.
- Zawsze warto zastanowić się nad tym, jak ustrukturalnić rozwiązanie i jaki będzie jego koszt.