

Programowanie zachłanne na przykładzie problemu wydawania reszty

Algorytm zachłanny

Określenie 1: Algorytm zachłanny charakteryzuje się tym, że w celu rozwiązania problemu w każdym kroku dokonuje ZACHŁANNEGO (najlepszego w danym momencie) wyboru rozwiązania podproblemu (rozwiązania częściowego). Jest to decyzja lokalnie optymalna.

Określenie 2: Algorytmy podejmujące w każdym kroku taką decyzję, która w danej chwili wydaje się najkorzystniejsza. Inaczej mówiąc, algorytmy zachłanne dokonują zawsze wyborów lokalnie optymalnych licząc, że doprowadzi to do znalezienia rozwiązania globalnie optymalnego. W ogólnym przypadku algorytmy zachłanne nie zawsze znajdują rozwiązanie optymalne. (źródło: algorytm.ency.pl)

Poprawność algorytmów zachłannych

- 1) Metody zachłanne NIE gwarantują najlepszego rozwiązania wszystkich problemów, dla których mogą być wykorzystane.
- 2) Jest grupa problemów, dla których metody zachłanne zawsze zwracają rozwiązanie optymalne, na przykład:
 - Algorytm Dijkstry - wyszukiwanie najkrótszych ścieżek w grafie,
 - Algorytm Kruskala - wyszukiwanie minimalnego drzewa rozpinającego.
- 3) W przypadku wydawania reszty można podać sytuacje, w których algorytm nie da rozwiązania najlepszego.

Algorytm zachłanny wydawania reszty

Problem: *Wydaj resztę używając jak najmniej sztuk dostępnych nominałów (monet, banknotów)*

Uwagi:

- Problem wydawania reszty może mieć odmiany w zależności od ustawienia zbioru nominałów, czyli odpowiedzi na pytania:
 - Czy zawsze można wydać resztę z dostępnych nominałów?
 - Czy ilość każdego nominału jest ograniczona, czy nie?
 - Czy zbiór nominałów jest uporządkowany?
- Metoda zachłanna wydawania reszty NIE gwarantuje w każdym przypadku rozwiązania optymalnego.

Kodowanie

1) Nasze założenia:

- tablica nominałów jest posortowana MALEJĄCO,
- nominały to liczby naturalne większe od 0,
- ilość sztuk każdego nominału NIE jest ograniczona,
- zawsze jest możliwe wydanie reszty do 0 (ostatni nominał to 1);
- reszta jest liczbą naturalną.

2) Przykładowy kod w C++:

```

/*
Zachłanny algorytm wydawania reszty.
Założenia:
  1) tablica nominałów posortowana MALEJĄCO
  2) ilość każdego nominału NIE jest ograniczona
  3) zawsze jest możliwe wydanie reszty do 0 (ostatni nominał to 1)
  4) reszta jest liczbą naturalną
*/

#include <iostream>
#include <string>
using namespace std;

int main(){
    //int nominal[]={20, 10, 2, 1}; // Tablica dostępnych nominałów
    int nominal[]={500, 200, 100, 200, 100, 50, 20, 10, 5, 2, 1}; // Tablica dostępnych nominałów
    int l_nominalow = sizeof(nominal) / sizeof(int); // Ilość elementów tablicy
    int reszta; // Reszta do wydania
    int ilosc = 0; // łączna ilość "monet" potrzebna do wydania reszty
    int pozycja = 0; // Pozycja aktualnego nominału z tablicy nominałów. Zaczynamy od początku (od największego nominału)
    int i, moneta;
    cout << "dostępne nominały (monety): \n";
    for(i=0; i<l_nominalow; i++) cout << nominal[i] << " ";
    cout << "\nreszta do wydania: "; cin >> reszta; //Wczytanie wartości reszty do wydania
    cout << endl;
    while(reszta>0){ //Dopóki mamy co wydawać, to wydajemy
        while(nominal[pozycja]>reszta) pozycja++; //Ustalenie największej monety <= od aktualnej reszty
        cout << "aktualna pozycja: " << pozycja;
        moneta=nominal[pozycja]; // Moneta - wartość nominału dla określonej pozycji
        cout << " moneta: " << moneta;
        cout << " ilość monet: " << reszta/moneta; // Ilość monet, które mieszczą się w reszcie
        ilosc=ilosc+reszta/moneta; // Obliczanie łącznej ilości monet potrzebnej do wydania reszty
        reszta=reszta%moneta; // Obliczenie nowej wartości reszty
        cout << " reszta: " << reszta << endl;
    }
    cout << "\ncalkowita ilosc potrzebnych monet do wydania reszty: " << ilosc;
    return 0;
}

```

Przykłady uruchomienia

1) Dla reszty 48 zł:

```

dostępne nominały (monety):
500 200 100 200 100 50 20 10 5 2 1
reszta do wydania: 48

aktualna pozycja: 6 moneta: 20 ilosc monet: 2 reszta: 8
aktualna pozycja: 8 moneta: 5 ilosc monet: 1 reszta: 3
aktualna pozycja: 9 moneta: 2 ilosc monet: 1 reszta: 1
aktualna pozycja: 10 moneta: 1 ilosc monet: 1 reszta: 0

calkowita ilosc potrzebnych monet do wydania reszty: 5

```

2) Dla reszty 4435 zł:

```
dostępne nominały (monety):
500 200 100 200 100 50 20 10 5 2 1
reszta do wydania: 4435

aktualna pozycja: 0 moneta: 500 ilosc monet: 8 reszta: 435
aktualna pozycja: 1 moneta: 200 ilosc monet: 2 reszta: 35
aktualna pozycja: 6 moneta: 20 ilosc monet: 1 reszta: 15
aktualna pozycja: 7 moneta: 10 ilosc monet: 1 reszta: 5
aktualna pozycja: 8 moneta: 5 ilosc monet: 1 reszta: 0
```

3) Zmieniony zbiór nominałów i reszta 9zł.

ROZWIĄZANIE NIEOPTYMALNE!!! MOŻNA WYDAĆ LEPIJ.

```
dostępne nominały (monety):
6 4 1
reszta do wydania: 9

aktualna pozycja: 0 moneta: 6 ilosc monet: 1 reszta: 3
aktualna pozycja: 2 moneta: 1 ilosc monet: 3 reszta: 0

calkowita ilosc potrzebnych monet do wydania reszty: 4
```

Metoda zachłanna mówi o 4 monetach potrzebnych do wydania reszty, a tymczasem wystarczą 3:

$$9 = 4 + 4 + 1$$

Wystarczą 3 monety!!!