

Automat wydaje resztę – dynamicznie wydawanie reszty



Jak automat wydaje resztę?

Automat kieruje się zasadą optymalizacji podczas wydawania reszty, polega ona na użyciu jak najmniejszej ilości wydanych monet.

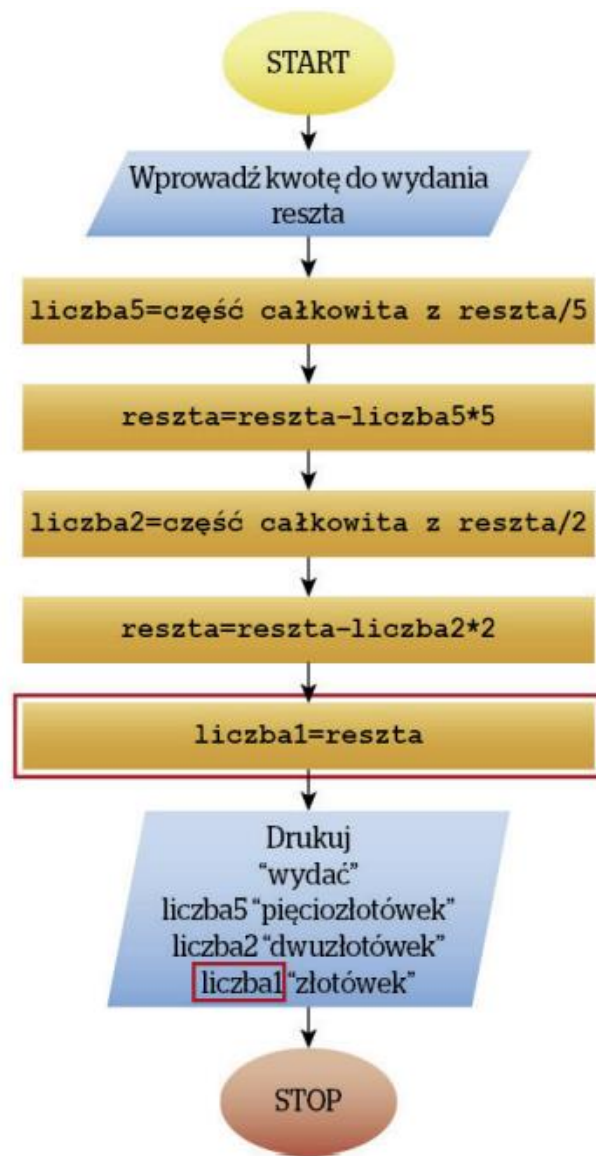
Automat wydaje resztę – dynamicznie wydawanie reszty



Wydawanie reszty największymi możliwie nominałami nazywamy **metodą zachłanną.**

Wydanie reszty w kwocie 13 złotych i 60 groszy przez automat odbędzie się (mając do dyspozycji wszystkie nominały w wystarczającej ilości) przez wydanie następujących nominałów:
5 zł+5zł+2zł+1zł+50gr+10gr.

Automat wydaje resztę – dynamicznie wydawanie reszty



Algorytm zachłanny wydający 6 zł. Algorytm wydaje od największego do najmniejszego nominału. Do dyspozycji są monety 5, 2 i 1 zł.

Gdybyśmy mieli do dyspozycji 5 i 2, to algorytm zachłanny wydałby nam tylko 5 zł.

Zmienne:

reszta – zawiera kwotę do wydania. W trakcie procesu wydawania zmienia wartość na równą kwocie, która pozostała do wydania.

liczba5, liczba2, liczba1 - zmienne, w których ostatecznie zostanie zapisana liczba poszczególnych nominałów do wydania.

Tą część algorytmu można pominąć. Zmienna reszta powyżej nie zmienia się do końca działania algorytmu.

Automat wydaje resztę – dynamicznie wydawanie reszty

Przed modyfikacją

```
#include <iostream>

using namespace std;

int liczba5 = 0, liczba2 = 0, liczba1 = 0, reszta;

int main()
{
    cout << "podaj kwote do wydania : ";
    cin >> reszta;

    liczba5 = reszta / 5;
    reszta = reszta - liczba5 * 5;
    liczba2 = reszta / 2;
    reszta = reszta - liczba2 * 2;
    liczba1 = reszta / 1;
    reszta = reszta - liczba1 * 1;

    cout << " do wydania "<< endl;
    cout << liczba5 << " szt. piecizlotowek" << endl;
    cout << liczba2 << " szt. dwuzlotowek" << endl;
    cout << liczba1 << " szt. zlotowek" << endl;

    return 0;
}
```

Po modyfikacji- optymalizacji

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

int liczba5 = 0, liczba2 = 0, liczba1 = 0, reszta;

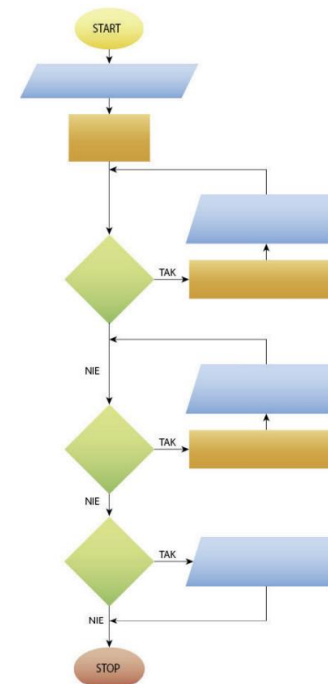
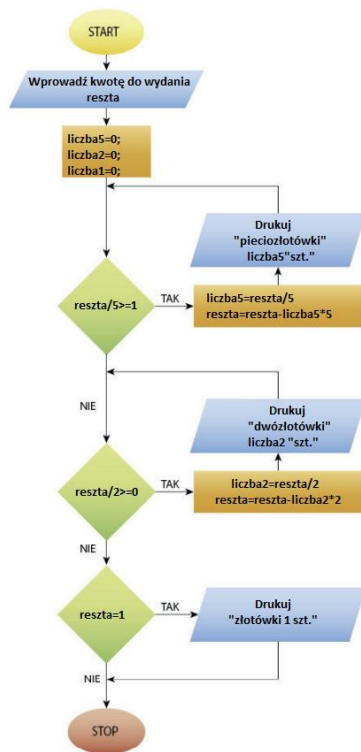
int main()
{
    cout << "podaj kwote do wydania : ";
    cin >> reszta;

    liczba5 = reszta / 5;
    reszta = reszta - liczba5 * 5;
    liczba2 = reszta / 2;
    reszta = reszta - liczba2 * 2;

    cout << " do wydania "<< endl;
    cout << liczba5 << " szt. piecizlotowek" << endl;
    cout << liczba2 << " szt. dwuzlotowek" << endl;
    cout << reszta << " szt. zlotowek" << endl;

    return 0;
} b
```

Dynamiczne wydawanie reszty bez wypisywania zerowych nominałów



```

#include <iostream>
using namespace std;

int liczba5 = 0, liczba2 = 0, liczba1 = 0, reszta;

int main()
{
    cout << "podaj kwote do wydania : ";
    cin >> reszta;
    cout << endl;
    if (reszta / 5 >=1)
    {
        liczba5 = reszta/5;
        reszta = reszta - liczba5 *5;
        cout << "piecizlotowki - " << liczba5 << "szt." << endl;
    }

    if (reszta / 2 >=1)
    {
        liczba2 = reszta/2;
        reszta = reszta - liczba2 *2;
        cout << "dwuzlotowki - " << liczba2 << "szt." << endl;
    }
    if (reszta ==1)
        cout << "zlotowki - 1 szt.";
}

```

Algorytm zachłanny bez zerowych reszt.

Dzięki zastosowaniu badania warunków można opuścić drukowanie i obliczenia dla nominałów mniejszych od pozostałej do wydania kwoty

Tablice - macierze

Zmienne tablicowe

Tablica (macierz) jest zmienną, która może przechowywać wiele wartości. Każda pozycja tablicy jest indeksowana (od 0 do n), co oznacza, że ma swój numer.

W związku z tym podczas deklaracji należy podać jej rozmiar.

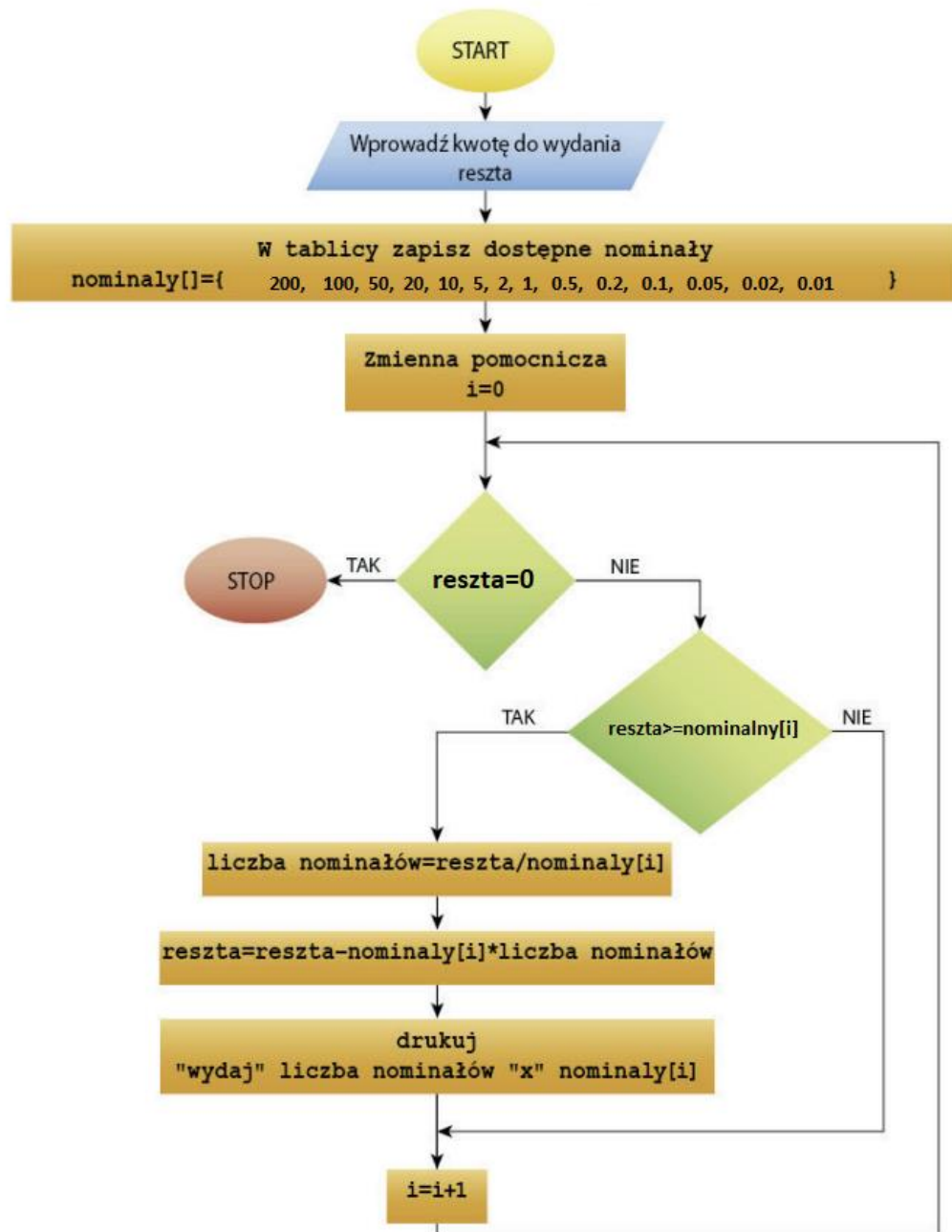
W C++, jeśli podczas deklaracji tablicy zostanie ona wypełniona danymi, podawanie rozmiaru nie jest konieczne. Podobnie jak inne zmienne, tablice muszą mieć nazwę i określony typ danych, jakie będą przechowywały.

Przykład deklaracji tablicy w C++:

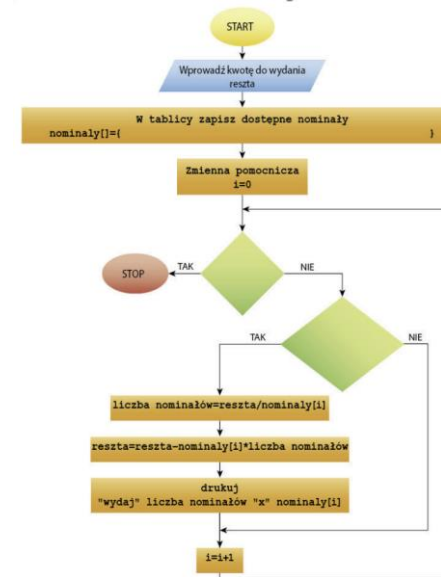
```
int tablica[]={20000,10000,5000,2000,1000,  
500,200,100,50,20,10,5,2,1};
```

lub inaczej

```
int tablica[13];
```

Automat wydaje resztę –
zastosowanie zmiennej tablicowej
– wszystkie nominały, z których
można wydawać resztę.



Krok	Działanie	Wartości
1.	Wprowadzenie kwoty do wydania	Reszta=40.10
2.	Wypełnienie tablicy nominałami	nominaly[13]={200,100,50,...,0.01}
3.	Indeks tablicy – zmienna i –ustawiony na pierwszej danej tablicy	i=0
4.	Czy reszta jest już wydana	Reszta=40.1 !=0 Kierunek – NIE
5.	Czy kwota do wydania zawiera w sobie przynajmniej jedną wartość nominału 200	nominaly[0]=200 40.1 < nominaly[0] Kierunek – NIE
6.	Indeks tablicy ustawiw się na następny nominał	i=0+1
7.	Czy reszta jest już wydana	reszta=40.1!=0 Kierunek- NIE
	Kroki 4,5,6,7 są wykonywane do momentu, gdy i=3	
8.	Reszta do wydania jest większa od 20	reszta>=nominaly[3]=20
9.	Obliczanie reszty dwudziestozłotówek	liczbanominalow=40.1/20=2 Nie ma części ułamkowej – zmienna jest typu int
10.	Obliczanie pozostałej kwoty po wydaniu 20 zł	reszta=40.1-(20*2)=0.1 zł
11.	Wydruk liczby 20 –złotówek	Wydaj 2x20
12.	Indeks następnego nominału	i=3+1=4
13.	Reszta jeszcze nie wydana	reszta=0.1!=0 Kierunek –NIE
	Dla nominałów od 4 do 9 wartość warunku reszta>=nominaly[i] – NIE	
14.	i=10	0.1>=nominaly[10]=0.1 Kierunek – TAK
15.	Obliczanie liczby 10-groszówek	liczbanominalow=0.1/0.1
16.	Obliczanie pozostałej kwoty po wydaniu 2x20 zł i 10 gr	Reszta=0.1-0.1*1=0
17.	Wydruk liczby dziesięciogroszówek	Wydaj 1x0.1
18.	Cała reszta wydana	Warunek reszta=0 spełniony koniec obliczeń

```

#include <iostream>
#include <math.h>
#include <stdlib.h>
using namespace std;

int liczbanominalow, i = 0;
float nominaly[] = {200,100,50,20,10,5,2,1,0.5,0.2,0.1,0.05,0.02,0.01}, reszta;

int main()
{
    cout << "Podaj kwote do wypłacenia: ";
    cin >> reszta;;

    cout << endl << "Wydaj nastepujace nominaly: " << endl << endl;

    while (reszta!=0)
    {
        if (reszta >= nominaly[i])
        {
            liczbanominalow=reszta/nominaly[i];
            reszta=round((reszta-(nominaly[i]*liczbanominalow))*100)/100;

            cout << nominaly[i] << "zl - " << liczbanominalow << "szt." << endl;

        }
        i++;
    }

    return 0;
}

```

Automat wydaje resztę – zastosowanie zmiennej tablicowej – wszystkie nominały, z których można wydawać resztę.