

## Scenariusz zajęć nr 5

### Temat: Kasjer

#### Cele:

W trakcie zajęć uczeń zapozna się z następującymi pojęciami:

- Używanie dwóch „równoległych” serii liczbowych,
- Prawidłowe uwzględnienie warunków brzegowych,
- Wykonywanie operacji całkowitoliczbowych na wielu elementach serii.

#### Wstęp:

Uruchomienie środowiska programistycznego, zapoznanie się z edytorem kodu źródłowego i sposobem kompilacji programu (kontynuacja).

#### *Dla nauczyciela:*

W razie braku zainstalowanego środowiska można skorzystać z dowolnego środowiska dostępnego w sieci Internet, na przykład `cpp.sh`.

#### Przebieg zajęć:

##### *Zapoznanie się z treścią zadania:*

Treść zadania jest dołączona do scenariusza zajęć w formie pliku `cpp_05-kasjer.pdf`.

#### *Dla nauczyciela:*

Tematem zajęć jest zadanie *Kasjer* dostępne na serwisie `szkopul.edu.pl`, wzorowane na zadaniu *Cashier* (1059A) z serwisu `codeforces.com`. Zadanie można rozwiązać w dowolnym języku programowania, a jego stopień trudności należy ocenić jako właściwy dla uczestników podstawowego kursu programowania.

## Wskazówki do rozwiązania zadania:

W zadaniu należy wyznaczyć długości kolejnych przerw pomiędzy klientami. Przerwa trwa od momentu zakończenia obsługi poprzedniego klienta (oznaczymy to przez  $t_p$ ), aż do rozpoczęcia obsługiwanego następnego klienta, czyli  $t_i$ ,  $i = 1, 2, \dots, n$ . Przechodząc do następnej przerwy nadajemy  $t_p$  wartość  $t_i + l_i$ .

Aby opis układu przerw był kompletny, musimy dodać jeszcze przerwę *przed* obsłużeniem pierwszego klienta (wystarczy na początku przyjąć  $t_p = 0$ ) oraz przerwę *po* obsłużeniu ostatniego klienta. Układ przerw przedstawia się zatem następująco:

- $t_p := 0$ ;
- Przerwa 1:  $(t_p, t_1)$   
Długość =  $t_1 - t_p$ ,  $t_p := t_1 + l_1$ ;
- Przerwa 2:  $(t_p, t_2)$   
Długość =  $t_2 - t_p$ ,  $t_p := t_2 + l_2$ ;
- Przerwa 3:  $(t_p, t_3)$   
Długość =  $t_3 - t_p$ ,  $t_p := t_3 + l_3$ ;  
...
- Przerwa  $n$ :  $(t_p, t_n)$   
Długość =  $t_n - t_p$ ,  $t_p := t_n + l_n$ ;
- Przerwa  $n + 1$ :  $(t_p, L)$   
Długość =  $L - t_p$ .

Używany przez nas symbol „:=” oznacza *przypisanie* (nadanie wartości).

Długości przerw podzielone przez czas jednego „dymka” (a dokładniej: *podłoga* z tego dzielenia) dają po zsumowaniu ilość wypalonych papierosów.

## Kod programu w języku C++:

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

int main()
{
    int t_p = 0;
    int n, L, a, n_a = 0, t, l;
    cin >> n >> L >> a;
    while(n-->0)
    {
        cin >> t >> l;
        n_a += (t - t_p) / a;
        t_p = t + l;
    }
    cout << n_a + (L - t_p) / a << endl;
    return 0;
}
```

Zmienna  $n_a$  (`n_a`) oznacza ilość wypalonych papierosów.

## Podsumowanie i uwagi końcowe:

Należy podkreślić, że nie można po prostu zsumować wszystkich przerw w pracy kasjera i „hurtem” podzielić przez czas na jeden papieros, ponieważ mamy do czynienia z operacjami całkowitoliczbowymi (dzielenie z podłogą). Część ułamkowa z każdej przerwy jest (oddzielnie) odrzucana.

Dane wejściowe (serie  $t_i$  oraz  $l_i$ ) nie muszą być przechowywane w tablicach – można przetwarzać je na bieżąco w trakcie wczytywania (*online*). Dane te należałoby przechować, gdyby były wykorzystywane ponownie w kodzie programu.