

# Obliczenia całkowitoliczbowe

---

W trakcie dzisiejszych ćwiczeń zajmiemy się obliczeniami całkowitoliczbowymi – zatem skoncentrujemy się na operacjach arytmetycznych na liczbach całkowitych (dla ustalenia uwagi: typu `int`).

W pierwszej części z pewnością przyda się przypomnienie operacji:

Dzielenia całkowitego: załatwia to operator `/` o ile oba argumenty są całkowite.

Reszty z dzielenia: tu do dyspozycji mamy operator `%`.

Prócz tego przyda się z pewnością instrukcja warunkowa.

# Zadanie rozgrzewkowe

---

Zadanie polega na podpowiedzeniu użytkownikowi, który chce przetransportować określoną liczbę bel słomy, ilu wozów (o określonej pojemności, wyrażonej w belach słomy) potrzebuje. Dodatkowo, program powinien podać, ile bel słomy można jeszcze przetransportować (o ile ostatni wóz nie jest całkiem wypełniony).

Na przykład, przy założeniu, że na wozie mieści się 7 bel słomy (tę wartość chciałbym móc łatwo zmienić), konsola programu może wyglądać tak:

Liczba bel słomy: **1**

Wozow: 1

Dodatkowych bel: 6

Wartość wprowadzona przez  
użytkownika

# Zadanie rozgrzewkowe – namysł

---

W zasadzie zadanie jest na tyle proste, że od razu można je napisać. Ale postanawiam na piśmie zostawić ślad, że chcę najpierw zrobić obliczenia, a potem wyświetlić wynik:

```
const int pojemnosc_wozu = 7;

int main()
{
    cout << "Liczba bel slomy: ";
    int liczba_bel;
    cin >> liczba_bel;

    // wyznaczenie liczby wozów i miejsca na dodatkowe bele

    cout << "   Liczba wozow: " << liczba_wozow << endl;
    cout << "   Miejsca na bele: " << wolne_miejsce << endl;
    return 0;
}
```

# Zadanie rozgrzewkowe – liczenie v1

---

W pierwszej wersji liczę „na szybko” liczbę wozów i wolne miejsce (iloraz całkowity i reszta z dzielenia przez pojemność wozu). Jeśli reszta z dzielenia jest niezerowa, to trzeba skorygować i liczbę wozów (jeden więcej) i wolne miejsce (bo mam liczbę bel na ostatnim wozie):

```
int liczba_wozow = liczba_bel / pojemnosc_wozu;  
int wolne_miejsce = liczba_bel % pojemnosc_wozu;  
if (wolne_miejsce != 0)  
{  
    liczba_wozow += 1;  
    wolne_miejsce = pojemnosc_wozu - wolne_miejsce;  
}
```

# Zadanie rozgrzewkowe – liczenie v2

---

Dość łatwo mogę pozbyć się korekcji liczby wozów „udając”, że chcę przewieźć nieco więcej bel słomy; dokładnie o pojemnosc\_wozu - 1:

```
int liczba_wozow = (liczba_bel + pojemnosc_wozu - 1)
                  / pojemnosc_wozu;
int wolne_miejsce = liczba_bel % pojemnosc_wozu;
if (wolne_miejsce != 0)
    wolne_miejsce = pojemnosc_wozu - wolne_miejsce;
```

Proszę przemyśleć tę pierwszą instrukcję – warto ten chwyt znać, choć miny jednej grupy na jej widok przekonały mnie, że trzeba zostawić wersję v1.

# Zadanie rozgrzewkowe – liczenie v3

---

W trakcie ćwiczeń padło pytanie, czy można wykonać to zadanie w ogóle bez instrukcji warunkowej. Oczywiście, pod warunkiem że znajdziemy sposób, żeby `pojemnosc_wozu - wolne_miejsce` w każdym przypadku sprowadzić do przedziału  $<0, pojemnosc\_wozu - 1>$ . To akurat stosunkowo proste:

```
int liczba_ozow = (liczba_bel + pojemnosc_wozu - 1)
                  / pojemnosc_wozu;
int wolne_miejsce = (pojemnosc_wozu - liczba_bel % pojemnosc_wozu)
                   % pojemnosc_wozu;
```

Ale mam teraz dwa wyrażenia, które przy czytaniu wymagają namysłu – zostają przy wersji v1.

## Bliżej zadania trzeciego

---

W zadaniu trzecim kluczem jest trójka jednostek wagi w systemie imperialnym:

kamień – funt – uncja

1 kamień = 14 funtów      1 funt = 16 uncji

Użytkownik podaje wagę w gramach (możemy spokojnie przyjąć, że jest to wartość całkowita).

Program wyświetla te podaną wagę w jednostkach imperialnych składając uncje, funty i kamienie. Możemy podać tutaj ograniczenia poszczególnych wartości:  $0 \leq \text{kamienie}$ ,  $0 \leq \text{funty} < 14$ ,  $0 \leq \text{uncje} < 16$ . Dodatkowo, program nie wyświetla tych jednostek, dla których trzeba byłoby wyświetlić 0.

# Parę przypadków testowych

---

Waga w gramach: **28**

uncji: 1

Waga w gramach: **453**

funtow: 1

Waga w gramach: **6350**

kamieni: 1

Waga w gramach: **10348**

kamieni: 1 funtow: 8 uncji: 13

Waga w gramach: **604**

funtow: 1 uncji: 5



## Dostarczanie rozwiązania

---

Rozwiązaniem zadania jest plik **zad\_m3.cpp** (dokładnie tak ma się nazywać). Proszę złożyć w Moodle tylko ten jeden plik.

Na początku pliku trzeba umieścić w komentarzu imię, nazwisko i nr albumu autora.

Pod nagłówkiem Tydzień 3 - 17-23.10 znajdziecie zadanie zatytułowane Kamienie – funty – uncje. W tym zadaniu każdy z Was powinien załadować plik `zad_m3.cpp` do:

**26 października 2022 23:59**