

Zadanie 4

Na podstawie materiału z rozdz. 9 (w szczególności temat „operatory bitowe”), proszę napisać i przetestować opisany niżej program:

- Program w pętli nieskończonej prosi użytkownika o wpisanie czterech liczb całkowitych nieujemnych i mniejszych od 256, które zapamiętuje w czteroelementowej tablicy `int`'ow (na przykład tablicy `col1`). Takie cztery liczby mogą na przykład reprezentować składowe *RGB* koloru i jego tzw. kanał α .
- Następnie program woła dla tej tablicy funkcję o deklaracji

```
int cols2int(int col1[]);
```

która liczy z podanej czteroelementowej tablicy „składa” w *jedną* liczbę typu `int` (na przykład `k`). Liczby z tablicy umieszczane są w kolejnych bajtach wyjściowej liczby, co jest możliwe, bo wszystkie są mniejsze od 256. Wynikową liczbę `k` funkcja zwraca, a w programie głównym ją drukujemy.
- Otrzymana w ten sposób liczba `k` jest następnie posyłana do funkcji o deklaracji

```
void int2cols(int k, int col2[]);
```

wraz z czteroelementową tablicą `int`'ow (tutaj nazwaną `col2`). Zadaniem tej funkcji jest rozłożyć otrzymaną liczbę `k` z powrotem na cztery liczby typu `int` z zakresu $[0, 255]$ równe co do wartości kolejnym czterem bajtom zmiennej `k`, i umieścić je w tablicy `col2`.
- Po powrocie z funkcji drukujemy cztery otrzymane liczby (powinniśmy odtworzyć wyjściową czwórkę liczb).
- Program kończy się, gdy użytkownik wpisze wszystkie cztery liczby równe 0.

Na przykład dla liczb 255,255,255,255 powinniśmy otrzymać `k = -1` i następnie odtworzyć tablicę składającą się z czterech wartości 255. Dla liczb 255,255,255,254 powinniśmy otrzymać `k = -2` lub `k = -16777217` w zależności od kolejności w jakiej „zapakowaliśmy” elementy tablicy do jednej liczby.

Termin: do 21 listopada (włącznie)

Rozwiązania, w postaci **jednego** pliku źródłowego zawierającego treść programu, proszę wrzucać w systemie EDU do katalogu „Foldery zadań / Zadanie_04”.

Nazwą pliku powinno być nazwisko z dużej litery (bez polskich znaków); rozszerzeniem musi być `.cpp`, czyli np. `Malinowska.cpp`.