

# Zadanie B – Podprogramy

Punktów do uzyskania: **10**

- Zadanie polega na implementacji zbioru przeładowanych podprogramów dodawania i mnożenia dowolnej ilości dowolnie dużych liczb całkowitych.
- Podprogramy o identyfikatorach Sum implementują dodawanie, zaś programy o identyfikatorach Mult implementują mnożenie.
- Ponadto podprogramy o identyfikatorach Operation implementują możliwość zastosowania w pojedynczym wywołaniu jednego z podprogramów Sum lub Mult przekazywanego w argumencie wraz z pozostałymi koniecznymi argumentami.
- Rozwiązanie musi być rozdzielone na trzy pliki źródłowe zgodnie z poniższą specyfikacją:
  - Plik źródłowy **Sum.cpp** obejmuje podprogramy o nagłówkach:
    - **string Sum ( int, const string\* );**
    - **string Sum ( int, ... );**
    - **void Sum ( string\*, int, const string\* );**
    - **void Sum ( string\*, int, ... );**
    - **void Sum ( string&, int, const string\* );**
    - **void Sum ( string&, int, ... );**
  - Plik źródłowy **Mult.cpp** obejmuje podprogramy o nagłówkach:
    - **string Mult ( int, const string\* );**
    - **string Mult ( int, ... );**
    - **void Mult ( string\*, int, const string\* );**
    - **void Mult ( string\*, int, ... );**
    - **void Mult ( string&, int, const string\* );**
    - **void Mult ( string&, int, ... );**
  - Plik źródłowy **Operation.cpp** obejmuje podprogramy o nagłówkach:
    - **string Operation ( string (\*)( int, const string\* ), int, const string\* );**
    - **string Operation ( string (\*)( int, const string\* ), int, ... );**
    - **void Operation ( string\*, string (\*)( int, const string\* ), int, ... );**
    - **void Operation ( string\*, string (\*)( int, const string\* ), int, const string\* );**
    - **void Operation ( string&, void (\*)( string\*, int, const string\* ), int, const string\* );**
    - **void Operation ( string&, void (\*)( string\*, int, const string\* ), int, ... );**
- W powyższej specyfikacji nagłówków:
  - Wartości zwracane przez podprogramy przekazują wyniki działań.
  - Argumenty całkowite oznaczają ilość argumentów działań i wynoszą nie mniej niż 2.
  - Pierwsze argumenty wskaźnikowe lub referencyjne przekazują wyniki działań.
  - Ostatnie argumenty wskaźnikowe przekazują argumenty działań.
  - Argumenty adresowe podprogramów odnoszą się do podprogramów Sum lub Mult.
- W kodzie źródłowym zabronione jest użycie pętli, implikując zakaz używania słów **while**, **for** oraz **goto**.
- Jedynymi dopuszczalnymi do włączania plikami nagłówkowymi są pliki **string** oraz **cstdint**.
- Rozwiązanie dopuszcza wyłącznie wiedzę przedstawioną na wykładzie, jednakże z wyłączeniem zmiennych globalnych oraz zmiennych statycznych.
- Pliki kodów źródłowych muszą w pierwszej linii zawierać komentarz z imieniem i nazwiskiem autora, zaś wysyłane na BaCę muszą być spakowane do pojedynczego pliku programem **Zip**.