

Ogólne wymagania i zalecenia dotyczące projektu pierwszego

1. Proszę przygotować implementację klasy realizującej treść zadania oraz program testujący. Program testujący powinien stworzyć dwa obiekty klasy z treści zadania oraz umożliwić użytkownikowi wykonanie operacji podanych w treści zadania na stworzonych obiektach. Do wyboru operacji do wykonania należy przygotować proste menu tekstowe. Należy pamiętać o sprawdzaniu poprawności danych podanych przez użytkownika. Jeżeli do stworzenia obiektu potrzebne są dodatkowe parametry, muszą być one również podane przez użytkownika (nie mogą być zapisane na stałe w kodzie programu testującego).
2. Każda przygotowywana klasa powinna zawierać konstruktor oraz, jeśli to potrzebne, destruktor.
3. Należy samodzielnie zaimplementować strukturę danych wskazaną w treści zadania – dopiero w ostatnim projekcie trzeba będzie wykorzystać elementy biblioteki STL.
4. Poza metodami realizującymi operacje wskazane w treści zadania przydatne może być także przygotowanie bardziej podstawowych metod, z których te pierwsze będą korzystać.
5. Operatory powinny być przeciążane w taki sposób, by wynik operacji zgodny był z jej definicją w konkretnej strukturze danych oraz z konwencją dotyczącą poszczególnych rodzajów operatorów (np. operator porównujący powinien zwracać wynik typu *bool*). Proszę zwrócić uwagę na możliwość wystąpienia błędów w tego typu operacjach – np. próba dodania dwóch wektorów o różnych wymiarach. W przypadku wykrycia błędu program powinien poinformować użytkownika o zaistniałej sytuacji.
6. Operacja „pobranie elementu” ze struktury danych (występująca w niektórych projektach) oznacza usunięcie wskazanego elementu i odpowiednią modyfikację pozostałych elementów tej struktury.
7. Wskazówka: struktury danych realizowane w projekcie 1 mają ustaloną wielkość lub wielkość maksymalną. Warto skorzystać z tego faktu przy pisaniu programów.

Projekt 1.10.

Proszę napisać klasę reprezentującą **wektor liczb rzeczywistych**. Liczba wymiarów wektora jest z góry ustalona przez użytkownika. Należy zaimplementować następujące operacje dla wektora:

1. Odczytanie wartości elementu o danej współrzędnej
2. Zmiana wartości elementu o danej współrzędnej
3. Wczytanie wektora (operator `>>`)
4. Dodawanie wektorów (operatory `+` i `+=`)
5. Odejmowanie wektorów (operatory `-` i `-=`)
6. Iloczyn skalarny wektorów (operator `*`)
7. Porównywanie wektorów (operatory `==` i `!=`)
8. Drukowanie wektora (operator `<<`)