

Zadanie 4

Wyspecjalizować szablon klasy `vec<T>` (zadanie 3) dla typu wskaźnikowego (`vec<T*>`). Obiekt klasy konkretyzowanej z szablonu ma posłużyć do przechowywania wskaźników do obiektów umieszczonych w pamięci dynamicznej. Należy pamiętać o odpowiednim zwolnieniu zasobów w chwili destrukcji obiektów klas utworzonych z tego szablonu.

Szablon klasy przetestować dla typu całkowitego i napisowego.

Przykład kodu testującego

```
vec<int*> t1(4);
for (unsigned i = 0; i < t1.size(); ++i)
    t1[i] = new int(i);
std::cout << t1 << '\n';

std::string* s[] = {new std::string("Ala"),
                    new std::string("ma"),
                    new std::string("kota")};

vec<std::string*> s1(s, s+3);
for (unsigned i = 0; i < 3; ++i)
    delete s[i];

*s1[2] = "psa";

std::cout << s1 << '\n';

const vec<int*> t2(t1);
std::cout << t2 << '\n';

vec<std::string*> s2(2);
s2 = s1;
std::cout << s2 << '\n';
```

Zadanie dodatkowe

Wyspecjalizować szablon klasy `vec<T>` (zadanie 3) dla typu logicznego (`bool`) uwzględniając przechowywanie wartości prawda/fałsz na pojedynczym bicie.