

Programowanie pod Windows

Zestaw 3, C#

23 marca 2003 roku

Stopień trudności zadania, między 1 (łatwe) a 3 (trudne), podany jest w nawiasie na początku treści każdego zadania.

1. (1) Przepisać, skompilować i uruchomić wszystkie programy z podrozdziału 3.1 (Podstawowe elementy języka C#) ze skryptu.
2. (1) Napisać program, który oblicza sumę, iloczyn i średnią z liczb podanych jako argumenty z linii poleceń.
3. (2) Napisać program, który wyznacza zbiór liczb pierwszych między 0 a 100000.
4. (2) Jeśli x_k jest pewnym przybliżeniem pierwiastka równania $f(x) = 0$, to lepszym przybliżeniem jest

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \quad (\text{wzór Newtona})$$

Korzystając z tego wzoru wyprowadzić algorytm obliczania $x^{\frac{1}{k}}$ i napisać program, który obliczy

- $27^{\frac{1}{3}}$
 - $100^{\frac{1}{4}}$
 - $17689^{\frac{1}{17}}$
5. (3) Napisać program wykonujący symboliczne obliczanie pochodnej. Wykorzystać rekurencyjne wzory:

$$\begin{aligned}(f + g)' &= f' + g' \\(f - g)' &= f' - g' \\(fg)' &= fg' + f'g \\ \left(\frac{f}{g}\right)' &= \frac{f'g - fg'}{g^2} \\(ax^n)' &= nax^{n-1}\end{aligned}$$

Na początek obliczać pochodne wielomianów, potem rozbudować program o funkcje trygonometryczne itp.

6. (1) Utworzyć klasę z konstruktorem bezparametrowym, który wypisuje jakiś komunikat. Przeciążyć metodę *ToString()* tak aby zwracała opis zawartości obiektu. Napisać program, który tworzy obiekt tej klasy oraz tablicę 100 elementów tej klasy. Za pomocą *foreach* przejrzeć wszystkie obiekty tej klasy i wypisać w oknie konsoli ich reprezentację napisową.
7. (1) Utworzyć klasę *COsoba* przechowującą takie informacje jak: imię, nazwisko, data urodzenia itp. Dodać odpowiednie konstruktory oraz przeciążyć metodę *ToString()*. Napisać program, który w tablicy pozwoli przechowywać do 100 obiektów tej klasy i dodać prosty interfejs użytkownika (w trybie tekstowym), pozwalający dodawać, usuwać, modyfikować, wyszukiwać elementy oraz tworzyć zestawienia posortowane po odpowiednich polach.
8. (2) Napisać klasę *CPolska* z dwuwymiarowym indeksem zwracającym nazwę miejscowości leżącej w odpowiednich współrzędnych geograficznych, tak aby klient klasy mógł napisać:

```
...
CPolska Polska = new CPolska();
string nazwaMiejscowosci = Polska[15.1, 52.4];
```

Indekser powinien przyjmować jako parametry liczby rzeczywiste i zwracać nazwę miejscowości leżącej najbliższej podanych współrzędnych. W wewnętrznych strukturach klasy przechować kilka przykładowych miejscowości.

9. (2) Napisać klasę do obsługi liczb zespolonych. Dodać odpowiednie konstruktory, przeciążyć odpowiednie operatory. Porównać wydajność obliczeń z użyciem zaprojektowanej klasy z obliczeniami przy użyciu klasy *complex* z C++.
10. (2) Napisać klasę *SuperString*, która będzie rozszerzać klasę *String* o operatory $<$, $<=$, $>$, $>=$. Dodać operatory konwersji (jawnej czy niejawniej?) między *String* a *SuperString*.
11. (2) Przedstawioną w skrypcie w rozdziale o strukturach strukturę RGB zamienić na klasę i przetestować w ten sposób wydajność wykorzystania klas i struktur. Jakie argumenty przemawiają za tym, aby jednak korzystać z typów referencyjnych, a nie tylko struktur?
12. (3) Zaprojektować klasę pozwalającą na operację na drzewach binarnych. Zaimplementować przechodzenie drzewa w szerz i w głąb. Napisać program, który wczytuje wyrażenie arytmetyczne napisane w ONP, buduje drzewo wyrażenia, a następnie przechodzi to drzewo obliczając wartość wyrażenia.

13. (1) Napisać program złożony z co najmniej 3 modułów.
14. (2) Napisać dwie wersje biblioteki zawierającej tę samą klasę, ale tak by implementacje przykładowych metod istotnie się różniły. W module głównym pozwolić użytkownikowi wybrać jedną z tych dwóch bibliotek, a następnie dynamicznie załadować bibliotekę, wykreować obiekt i wywołać jego metody oraz przechwycić wyniki.
15. (2) Napisać program z wykorzystaniem kodu niebezpiecznego, który będzie kopiował dane między tablicami. Następnie napisać program, który kopiuje tablice za pomocą pętli *for*. Porównać prędkość działania programu ze wskaźnikami i bez.
16. (2) Poszukać w sieci lub napisać własny inny niż przedstawiony w skrypcie arkusz stylów XSL dla formatowania dokumentacji kodu. Efekt zastosowania arkusza stylów obejrzyć w przeglądarce.