



KOŁOKWIUM „UŻYTKOWANIE KOMPUTERÓW”

Wykonane zadanie należy skompresować (np.: zip, rar) i umieścić w katalogu *FTP/wkaminski/studenci*. Nazwa pliku powinna być zgodna ze schematem:

WID_yyy_UKO_sxxxx.zip

(gdzie *yyy* jest numerem grupy studenckiej, a *xxxx* numerem indeksu)

UWAGA: NALEŻY WYBRAĆ DO WYKONANIA **JEDNO** Z PONIŻSZYCH ZADAŃ

Zadanie 1 (20 pkt.)

Należy narysować wykres liniowy. Pierwszym krokiem jest narysowanie układu współrzędnych w taki sposób, aby środek układu (a więc punkt 0,0) znajdował się na środku okienka. W przypadku dokonania zmiany rozmiaru okienka przez użytkownika układ współrzędnych powinien się dostosować. Dostarczono plik tekstowy *data.file*, który zawiera listę punktów stanowiących kluczowe elementy wykresu. Punkty są zorganizowane w następujący sposób:

$x_0, y_0; x_1, y_1; \dots; x_i, y_i$

(współrzędna *x*, kropka, współrzędna *y*) Liczba punktów nie jest z góry określona. Przykładowo zestaw punktów mógłby wyglądać tak:

-2.0;-1.7;-1.1;-1.0;0.1;1.1;2.5

Jeżeli to możliwe – wykres należy wygładzić (rysujemy krzywymi). Zakres wartości na osiach *X* i *Y* powinien dostosować się do danych.

UWAGA: Podpowiedź do zadania nr 1 znajduje się na drugiej stronie dokumentu

Zadanie 2 (15 pkt.)

Należy napisać aplikację, która narysuje prostokąt o wymiarach 40x40. Środek prostokąta powinien zostać połączony linią ze środkiem okienka. Początkowa pozycja prostokąta to środek okienka. Należy dać użytkownikowi możliwość przesuwania narysowanego prostokąta przy użyciu myszki (Drag&Drop). Podczas przesuwania prostokąt powinien być nadal połączony linią ze środkiem okienka. Dodatkowo należy obliczyć odległość (w pikselach) od prostokąta do środka okienka i wyświetlić ją przy użyciu komponentu Label (wartość ta powinna być aktualna w każdym momencie działania programu). Dodatkowo podwójne kliknięcie powinno resetować aplikację, czyli ustawiać prostokąt i linię w położeniu początkowym

Zadanie 3 (10 pkt.)

Należy napisać kalkulator. Kalkulator powinien być zbudowany ze zbioru przycisków umożliwiających:

- wprowadzenie liczb;
- przeprowadzenie operacji na wprowadzonych liczbach:
 - dodawanie
 - odejmowanie
 - mnożenie
 - dzielenie
 - potęgowanie
 - pierwiastkowanie
- zmianę znaku (+/-)
- odczytanie wyniku

Ponadto kalkulator powinien mieć wbudowany mechanizm pamięci działający na zasadzie sumowania (MC – Memory Clear, MR – Memory Release, M+ – Memory Add). Zasada działania pamięci dokładnie taka, jak w przypadku kalkulatora dostarczanego wraz z systemem Windows (pomijając dodatkowe funkcje jak: M-, MS).

```
//import bibliotek
using System.IO;
....
//wczytywanie pliku plik.txt o zawartości 1;2;3;4
//otwieramy strumień wczytujący plik.txt
//plik ten powinien znajdować się tam, gdzie znajduje się plik .exe naszej aplikacji
TextReader tr = new StreamReader("plik.txt");
//odczytujemy pojedynczą linię tekstu z pliku i zapisujemy w postaci zmiennej string
string linia = tr.ReadLine();
//zamykamy strumień wejściowy
tr.Close();
//skoro wczytany string ma postać 1;2;3;4 musimy wydobyć poszczególne wartości
//dzielimy ten napis na mniejsze napisy względem podanego znaku ;
//wynikiem jest tablica napisów
string[] tablica = linia.Split(';');
//zawartość tablicy możemy wykorzystać na wiele sposobów
//zapisywanie pliku - mam nadzieję, że nie wymaga komentarza
TextWriter tw = new StreamWriter("result.txt");
tw.WriteLine(„linia tekstu do zapisu”);
tw.Close();
```

UWAGA: Maksymalna liczba punktów możliwych do zdobycia to 15.