

## Funkcja sklejana stopnia trzeciego z użyciem macierzy trójdzielnej.

W ogólności przez funkcję sklejaną rozumie się każdą funkcję przedziałami wielomianową. Nas będą jednak interesować szczególne funkcje tego typu i dlatego termin **funkcje sklepane** zarezerwujemy dla funkcji przedziałami wielomianowych i posiadających dodatkowe własności.

Do wykonania zadania skorzystałem ze wzoru:

$$p(x) = p_j + \left[ \frac{p_{j+1} - p_j}{h} - \frac{1}{6}hp_{j+1}'' - \frac{1}{3}hp_j'' \right](x - x_j) + \frac{p_j''}{2}(x - x_j)^2 + \frac{p_{j+1}'' - p_j''}{6h}(x - x_j)^3$$

$$p_j \equiv f(x_j)$$

Następnie  $d^2p/dx^2$  (for  $j = 1, \dots, n$ ) może zostać obliczone korzystając z macierzy trójdzielnej.

Skorzystałem z programu z zeszytów laboratoriów do obliczenia macierzy trójdzielnej. Program zoptymalizowałem i nieco przerobiłem, żeby przyjemniej się korzystało, a kod podzieliłem na funkcje, wcześniej uznałem, że nie jest to konieczne. Program powiększył się i teraz jest bardziej przejrzysty.

W programie na wstępie dostajemy informacje, że przygotowane zostały 2 zestawy danych, na których program został przetestowany, ale oczywiście nie ma ograniczenia do rozbudowy o własne funkcje, a zestawy danych są prawie dowolne. Muszą się mieścić w zakresie typu double. Jednak do testów należy wybrać jeden z przygotowanych zestawów i wpisać go z klawiatury zgodnie z instrukcją. Następnie dla wybranego zestawu należy wpisać pierwszą liczbę od lewej strony. Przygotowane zostały również 2 funkcje do wyboru, na których można przetestować program.  $f(x) = \sin(x)$  oraz  $f(x) = 1/(1+x^2)$  następnie program oblicza wartości  $x$  korzystając z macierzy trójdzielnej.

Przykładowe zestawy testowe prezentują się następująco:

$x = \{-6, -5, -3, -2, 0, 1.5, 3, 4, 6, 7, 8\}$ ,  $h = \{1, 2, 1, 2, 1.5, 1.5, 1, 2, 1, 1\}$

$x = \{-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5\}$ ,  $h = \{1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1\}$

Na koniec działania programu wyświetlane są wzory na funkcje sklepane stopnia trzeciego.

Poniżej zamieszczam 2 zrzuty z ekranu dla 2 różnych przykładowych danych, są one również dołączone w osobnym pliku dla lepszej widoczności.:

```

Konsola debugowania programu Microsoft Visual Studio

1. Dla zestawu wezlow x=[-6, -5, -3, -2, 0, 1.5, 3, 4, 6, 7, 8]
krok h wynosi:
1 2, 1, 2, 1.5, 1.5, 1, 2, 1, 1

2. Drugi zestaw dla ktorego bedziemy testowac program to x=[-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5]
krok h wynosi:
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1

Prosze wybrac zestaw testowy, a nastepnie wpisac jego krok z klawiatury
1
1
1
1.5
1.5
1
2
1

uprowadz pierwsza wartosc zgodnie z ustalonym przez ciebie krokiem: -6

Do przetestowania programu przygotowalem 2 funkcje
f(1) f(x) = 1.0 / (1.0 + x * x)
druga z nich to
f(2) f(x) = sin(x)
wybierz na ktorej przetestowac program 1/2 jesli wpiszesz inna liczbe calkowita program wykona sie dla f(x) = sin(x)
7
-6 -5 -3 -2 0 1.5 3 4 6 7 8
0 -0.0773 0.095601 10.0648 0.264593 -7.41540 -0.16019 0.25186 1.37705 -3.62418 0
0 0 2 1 2 1.5 1.5 1 2 1 0
1 4 4 4 4 4 4 4 4 1
0 1 2 1 1.5 1.5 1 2 0 0

Obliczona beta z naszych danych 1 do n:
0 -0.25 -0.206667 -0.207857 -0.207943 -0.207940 -0.207940 -0.207940 -0.207940 -0.207940 0 0

Obliczona gamma z naszych danych od 1 do n:
0 -2.66933 0.864568 2.46972 -0.626295 -1.15083 0.280285 2.13508 -0.387044 -0.867174 0

Nazwe x w kolejnosci: p1,p1-1, ... p33,p32,p31
0 -0.867174 -0.155485 2.17764 -0.387212 -1.07550 -0.3381 2.56028 0.161826 -2.70979 0
2
3
4
5
6
7
8
9

Dane do 'a' 0 0.173852 -1.14743 -0.675344 1.05942 1.0489 -0.928515 -1.15981 0.984163 1.89262 0
Dane do 'b' 0 -0.435887 -0.077427 1.08882 -0.151606 -0.537793 -0.16096 1.28814 0.889132 -1.35489 0
Dane do 'c' 0 0.118015 0.194427 -0.413475 -0.0643645 0.0819429 0.322042 -0.399742 -0.239381 0.451631 0

1. p(x) = 0(x + (-6)) + 0(x + (-6))^2 + 0(x + (-6))^3
2. p(x) = 0.173852(x + (-5)) + -0.435887(x + (-5))^2 + 0.118015(x + (-5))^3
3. p(x) = -1.14743(x + (-3)) + -0.077427(x + (-3))^2 + 0.194427(x + (-3))^3
4. p(x) = -0.675344(x + (-2)) + 1.08882(x + (-2))^2 + -0.413475(x + (-2))^3
5. p(x) = 1.05942(x + (0)) + -0.151606(x + (0))^2 + -0.0643645(x + (0))^3
6. p(x) = 1.0489(x + (1.5)) + -0.537793(x + (1.5))^2 + 0.0819429(x + (1.5))^3
7. p(x) = -0.928515(x + (3)) + -0.16096(x + (3))^2 + 0.322042(x + (3))^3
8. p(x) = -1.15981(x + (4)) + 1.28814(x + (4))^2 + -0.399742(x + (4))^3
9. p(x) = -0.984163(x + (6)) + 0.889132(x + (6))^2 + 0.239381(x + (6))^3
10. p(x) = 1.89262(x + (7)) + -1.35489(x + (7))^2 + 0.451631(x + (7))^3
11. p(x) = 0(x + (8)) + 0(x + (8))^2 + 0(x + (8))^3

C:\Users\Adas\Desktop\Obliczenia_Symboliczne\laboratoria\Obliczenia_Symboliczne_3\Debug\Obliczenia_Symboliczne_3.exe (proces 9392) zakończono z kodem 0.
Aby automatycznie zamknąć konsolę po zatrzymaniu debugowania, włóż opcję Narzędzia -> Opcje -> Debugowanie -> Automatycznie zamknij konsolę po zatrzymaniu debugowania.
Naciśnij dowolny klawisz, aby zamknąć to okno...
```

```

Konsola debugowania programu Microsoft Visual Studio

1. Dla zestawu wezlow x=[-6, -5, -3, -2, 0, 1.5, 3, 4, 6, 7, 8]
krok h wynosi:
1 2, 1, 2, 1.5, 1.5, 1, 2, 1, 1

2. Drugi zestaw dla ktorego bedziemy testowac program to x=[-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5]
krok h wynosi:
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1

Prosze wybrac zestaw testowy, a nastepnie wpisac jego krok z klawiatury
1
1
1
1
1
1
1
1

uprowadz pierwsza wartosc zgodnie z ustalonym przez ciebie krokiem: -5

Do przetestowania programu przygotowalem 2 funkcje
f(1) f(x) = 1.0 / (1.0 + x * x)
druga z nich to
f(2) f(x) = sin(x)
wybierz na ktorej przetestowac program 1/2 jesli wpiszesz inna liczbe calkowita program wykona sie dla f(x) = sin(x)
7
-5 -4 -3 -2 -1 0 1 2 3 4 5
0 0.124887 0.352941 1.2 1.2 -6 1.2 1.2 0.352941 0.124887 0
0 0 1 1 1 1 1 1 1 0
1 4 4 4 4 4 4 4 4 1
0 1 1 1 1 1 1 1 0 0

Obliczona beta z naszych danych 1 do n:
0 -0.25 -0.206667 -0.207857 -0.207943 -0.207940 -0.207940 -0.207940 -0.207940 -0.207940 0 0

Obliczona gamma z naszych danych od 1 do n:
0 0.031217 0.8057719 0.298449 0.241564 -1.07242 0.709662 0.115309 0.0036734 0.0164801 0

Nazwe x w kolejnosci: p1,p1-1, ... p33,p32,p31
0 0.0164801 0.0592785 0.099425 0.743021 -1.07151 0.743021 0.899425 0.0592785 0.0164801 0
2
3
4
5
6
7
8
9

Dane do 'a' 0 0.0846529 0.16367 0.343021 1.06424 1 -0.0642446 0.0509786 0.0363303 0.0329942 0
Dane do 'b' 0 0.00820184 0.0296393 0.0497125 0.371511 -0.935755 0.371511 0.0497125 0.0296393 0.00820184 0
Dane do 'c' 0 0.00714607 0.00669108 0.107266 -0.435755 0.435755 -0.107266 -0.00669108 -0.00714607 -0.00273368 0

1. p(x) = 0(x + (-5)) + 0(x + (-5))^2 + 0(x + (-5))^3
2. p(x) = 0.0846529(x + (-4)) + 0.00820184(x + (-4))^2 + 0.00714607(x + (-4))^3
3. p(x) = 0.16367(x + (-3)) + 0.0296393(x + (-3))^2 + 0.00669108(x + (-3))^3
4. p(x) = 0.343021(x + (-2)) + 0.0497125(x + (-2))^2 + 0.107266(x + (-2))^3
5. p(x) = -1.06424(x + (0)) + 0.371511(x + (0))^2 + -0.435755(x + (0))^3
6. p(x) = 1(x + (0)) + -0.035755(x + (0))^2 + 0.435755(x + (0))^3
7. p(x) = -0.0042446(x + (1)) + 0.371511(x + (1))^2 + -0.107266(x + (1))^3
8. p(x) = 0.0509786(x + (2)) + 0.0497125(x + (2))^2 + -0.00669108(x + (2))^3
9. p(x) = 0.0263303(x + (3)) + 0.0296393(x + (3))^2 + -0.00714607(x + (3))^3
10. p(x) = -0.0320184(x + (4)) + 0.00820184(x + (4))^2 + -0.00273368(x + (4))^3
11. p(x) = 0(x + (5)) + 0(x + (5))^2 + 0(x + (5))^3

C:\Users\Adas\Desktop\Obliczenia_Symboliczne\laboratoria\Obliczenia_Symboliczne_3\Debug\Obliczenia_Symboliczne_3.exe (proces 14220) zakończono z kodem 0.
Aby automatycznie zamknąć konsolę po zatrzymaniu debugowania, włóż opcję Narzędzia -> Opcje -> Debugowanie -> Automatycznie zamknij konsolę po zatrzymaniu debugowania.
Naciśnij dowolny klawisz, aby zamknąć to okno...
```

Wykonał

Adam Majchrzak s176708 21.05.2020