Sprawozdanie projekt 1

Temat projektu:

Obliczanie całek złożoną kwadraturą prostokątów

Matematyczny opis metody

A screenshot of a graph

Description automatically generated

Rysunek 1: Fragment notatek z wykładu tłumaczący złożoną metodę prostokątów

A math equations and formulas written on a white board

Description automatically generated

Rysunek 2: Wyprowadzenie wzoru na obliczanie złożonej kwadratury prostokątów, użytego w implementacji funkcji

Opis programu

Program zawiera funkcję *przyblizenieCalki*, która przyjmuje cztery argumenty:

* *a* – początek przedziału, po którym całkujemy
* *b* – koniec przedziału, po którym całkujemy
* *f* – funkcja, którą całkujemy
* *e* – oczekiwana dokładność wartości całki (oczekiwana maksymalna wartość bezwględnej różnicy kolejnych przybliżeń),

a zwraca *w* – przybliżoną wartość całki obliczoną złożoną kwadraturą prostokątów z dokładnością e.

Funkcja *przyblizenieCalki* wykorzystuje funkcję pomocniczą *przyblizenieDlaM*, która przyjmuje trzy argumenty:

* *Xm* – wektor xk (k[m]) końców m podprzedziałów przedziału [*a*; *b*]
* *f* – funkcja, którą całkujemy
* *Hm* – długość podprzedziału [xk-1; xk] (k[m]),

a zwraca wektor dwuelementowy zawierający:

* *Sm* - przybliżoną wartość całki obliczoną złożoną kwadraturą prostokątów, po podzieleniu przedziału [*a*; *b*] na m podprzedziałów
* *X2m* – wektor xk (k[2m]) końców 2m podprzedziałów przedziału [*a*; *b*].

Działanie funkcji *przyblizenieDlaM*:

1. Wyznacza wartość wektora *Am*, którego elementy to ak (k[m]), czyli środki przedziałów [xk-1; xk] (k[m]), odejmując od wektora *Xm* wartość .
2. Wyznacza wartość wektora *Pm*, którego elementy to *f*(ak) (k[m]).
3. Wyznacza wartość *Sm*, sumując elementy wektora *Pm* i dodając do sumy *Hm*.
4. Wyznacza wartość wektora *X2m*, którego elementy to wszystkie wartości wektora *Xm* i wektora *Am*.
5. Zwraca *Sm* i wektor *X2m*.

Działanie funkcji *przyblizenieCalki*:

1. Ustala *m = 1*, *Xm = [b]*, *Hm =* .
2. Ustala *m =* .
3. Wyznacza *Sm2*, czyli przybliżenie całki dla podprzedziałów i nowe *Xm*, wykorzystując funkcję pomocniczą *przyblizenieDlaM(Xm, f, Hm)*.
4. Ustalamy nowe *Hm = .*
5. Wyznacza *Sm*, czyli przybliżenie całki dla *m* podprzedziałów i nowe *Xm*, znowu wykorzystując funkcję pomocniczą *przyblizenieDlaM(Xm, f, Hm)*.
6. Powtarza poniższe kroki, dopóki przybliżenie całki nie jest wystarczająco dokładne

(tzn.: *|Sm – Sm2|>e):*

* 1. Ustala *m =* .
  2. Ustala *Sm2 = Sm.*
  3. Wyznacza *Sm*, czyli przybliżenie całki dla *m* podprzedziałów i nowe *Xm*, znowu wykorzystując funkcję pomocniczą *przyblizenieDlaM(Xm, f, Hm)*.

1. Gdy przybliżona wartość całki jest wystarczająco dokładna, czyli to funkcja zwraca *Sm.*

Przykłady

Przyjmujemy oznaczenia:

* m – liczba podprzedziałów wykorzystana do obliczenia przybliżenia całki
* e – bezwzględna różnica wartości kolejnych przybliżeń Sm i Sm/2

Jeżeli w tabeli nazwa kolumny kończy się „\_Z01” oznacza to, że ta kolumna zawiera wartości obliczone dla całki z f(x) po przedziale [0;1], a jeżeli kończy się „\_Z56” – dotyczy całki z f(x) po [5;6]

A graph of a graph

Description automatically generated

Rysunek 3: Wykres funkcji sin(x) i jej interpolacji wielomianem stopnia 0 dla 32 węzłów na przedziale [0;1]

Na wykresie (Rysunek 3) widać, że funkcja interpolująca przyjmuje bardzo zbliżone wartości do funkcji .

Tabela 1: Wynik i błędy implementowanej funkcji dla f(x) = sin(x) na przedziale [0;1] dla różnych eA table with numbers and letters

Description automatically generated

A graph with a line

Description automatically generated

Rysunek 4: Wykres dokładnej wartości całki z sin(x) po [0;1] i jej przybliżenia w zależności od m

Tabela 2: Wynik i błędy implementowanej funkcji dla f(x) = sin(x) na przedziale [0;1] dla pierwszych 10 wartości m

A table of numbers with black text

Description automatically generated

Z tabel (Tabela 1, Tabela 2) i wykresu (Rysunek 4) wynika, że całka jest dobrze przybliżona zaimplementowaną funkcją już dla względnie małych m.

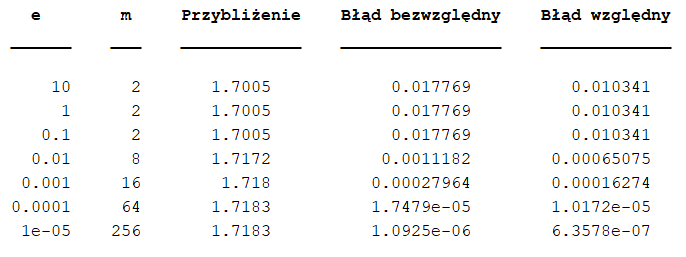
A graph with a line

Description automatically generated

Rysunek 5: Wykres funkcji exp(x) i jej interpolacji wielomianem stopnia 0 dla 32 węzłów na przedziale [0;1]

Na wykresie (Rysunek 5) widać, że funkcja interpolująca przyjmuje bardzo zbliżone wartości do funkcji .

Tabela 3: Wynik i błędy implementowanej funkcji dla f(x) = exp(x) na przedziale [0;1] dla różnych e



A white background with a blue line

Description automatically generated

Rysunek 6: Wykres dokładnej wartości całki z exp(x) po [0;1] i jej przybliżenia w zależności od m

Tabela 4: Wynik i błędy implementowanej funkcji dla f(x) = exp(x) na przedziale [0;1] dla pierwszych 10 wartości m

A screenshot of a table

Description automatically generated

Z tabel (Tabela 3, Tabela 4) i wykresu (Rysunek 6) wynika, że całka jest dobrze przybliżona już dla względnie małych m.

A graph of a curve

Description automatically generated with medium confidence

Rysunek 7: Wykres funkcji ln(x) i jej interpolacji wielomianem stopnia 0 dla 32 węzłów na przedziale [0;1]

Na wykresie (Rysunek 7) widać, że funkcja interpolująca przyjmuje tym bardziej zbliżone wartości do funkcji im x jest większe. Największa różnica w wartościach funkcji a wartościach funkcji interpolacyjnej jest w okolicy zera.

A graph with a line

Description automatically generated

Rysunek 8: Wykres funkcji ln(x) i jej interpolacji wielomianem stopnia 0 dla 32 węzłów na przedziale [5;6]

Na wykresie (Rysunek 8) widać, że funkcja interpolująca przyjmuje bardzo zbliżone wartości do funkcji na tym przedziale.

1. Porównanie wyników dla przybliżania całek i :

Tabela 5: Wynik i błędy implementowanej funkcji dla f(x) = ln(x) na przedziałach: [0;1] i [5;6] dla rożnych e

A screenshot of a computer

Description automatically generated

A graph with a red line and blue line

Description automatically generated

Rysunek 9: Wykres błędu względnego przybliżenia całki z ln(x) po przedziałach [0;1] i [5;6] w zależności od m

Tabela 6: Wynik i błędy implementowanej funkcji dla f(x) = ln(x) na przedziałach: [0;1] i [5;6] dla pierwszych 10 wartości m

A screenshot of a computer

Description automatically generated

Z tabel (Tabela 5, Tabela 6) i wykresu (Rysunek 9) wynika, że całka jest o wiele lepiej przybliżona od całki .

A graph with a curved line

Description automatically generated

Rysunek 10: Wykres funkcji ln(x) na przedziale [0;6]

Jest to spowodowane tym, że funkcja jest bardziej gładka, mniej zmienna i wolniej rośnie na przedziale [5;6] niż na przedziale [0;1], co widać na wykresie (Rysunek 10). Funkcja ma gwałtowny skok przy zerze, co wpływa na niedokładność jej przybliżenia.

A graph with a line

Description automatically generated

Rysunek 11: Wykres funkcji sin(1/x) i jej interpolacji wielomianem stopnia 0 dla 32 węzłów na przedziale [0;1]

Na wykresie (Rysunek 11) widać, że funkcja interpolująca przyjmuje tym bardziej zbliżone wartości do funkcji im x jest większe. Największa różnica w wartościach funkcji funkcji interpolacyjnej jest w okolicy zera, gdzie funkcja m gwałtowe i duże wahania wartości.

A graph with lines and numbers

Description automatically generated

Rysunek 12: Wykres funkcji sin(1/x) i jej interpolacji wielomianem stopnia 0 dla 32 węzłów na przedziale [5;6]

Na wykresie (Rysunek 12) widać, że funkcja interpolująca przyjmuje bardzo zbliżone wartości do funkcji na tym przedziale.

1. Porównanie wyników przybliżania całek i :

Tabela 7: Wynik i błędy implementowanej funkcji dla f(x) = sin(1/x) na przedziałach: [0;1] i [5;6] dla rożnych e

A screenshot of a computer

Description automatically generated

A graph with red and blue lines

Description automatically generated

Rysunek 13: Wykres błędu względnego przybliżenia całki z sin(1/x) po przedziałach [0;1] i [5;6] w zależności od m

Tabela 8: Wynik i błędy implementowanej funkcji dla f(x) = sin(1/x) na przedziałach: [0;1] i [5;6] dla pierwszych 10 wartości m

A screenshot of a computer

Description automatically generated

Z tabel (Tabela 7, Tabela 8) i wykresu (Rysunek 13) wynika, że całka jest o wiele lepiej przybliżona od całki . Jest to spowodowane tym, że funkcja jest bardziej gładka i mniej zmienna na przedziale [5;6] niż na przedziale [0;1], co widać na wykresie poniżej (Rysunek 14). Na przedziale [0;1] funkcja ma bardzo duże wahania wartości (blisko zera), przez co nawet zwiększając liczbę przedziałów trudno jest zwiększyć dokładność.

A graph of a function

Description automatically generated

Rysunek 14: Wykres funkcji sin(1/x) na przedziale [0;6]

A graph with lines and numbers

Description automatically generated

Rysunek 15: Wykres funkcji sin(1/x)\*ln(1/x) i jej interpolacji wielomianem stopnia 0 dla 32 węzłów na przedziale [0;1]

Na wykresie (Rysunek 15) widać, że funkcja interpolująca przyjmuje tym bardziej zbliżone wartości do funkcji im x jest większe. Największa różnica w wartościach funkcji funkcji interpolacyjnej jest w okolicy zera, gdzie funkcja ma gwałtowe i duże wahania wartości.

A graph with a line

Description automatically generated

Rysunek 16: Wykres funkcji sin(1/x)\*ln(1/x) i jej interpolacji wielomianem stopnia 0 dla 32 węzłów na przedziale [5;6]

Na wykresie (Rysunek 16) widać, że funkcja interpolująca przyjmuje bardzo zbliżone wartości do funkcji na tym przedziale.

1. Porównanie wyników przybliżania całek:

i :

Tabela 9: Wynik i błędy implementowanej funkcji dla f(x) = sin(1/x)\*ln(1/x) na przedziałach: [0;1] i [5;6] dla rożnych e

A screenshot of a computer

Description automatically generated

A graph with a line and a red line

Description automatically generated

Rysunek 17: Wykres błędu względnego przybliżenia całki z sin(1/x)\*ln(1/x) po przedziałach [0;1] i [5;6] w zależności od m

Tabela 10: Wynik i błędy implementowanej funkcji dla f(x) = sin(1/x)\*ln(1/x) na przedziałach: [0;1] i [5;6] dla pierwszych 10 wartości m

A screenshot of a computer

Description automatically generated

Z tabel (Tabela 9, Tabela 10) i wykresu (Rysunek 17) wynika, że całka jest o wiele lepiej przybliżona od całki . Jest to spowodowane tym, że funkcja

jest bardziej gładka i mniej zmienna na przedziale [5;6] niż na przedziale [0;1], co widać na wykresie poniżej (Rysunek 18). Na przedziale [0;1] funkcja ma bardzo duże wahania wartości (blisko zera), przez co nawet zwiększając liczbę przedziałów trudno jest zwiększyć dokładność.

A line graph with numbers

Description automatically generated

Rysunek 18: Wykres funkcji sin(1/x)\*ln(1/x) na przedziale [0;6]

A graph with numbers and lines

Description automatically generated

Rysunek 19: Wykres funkcji x^21 i jej interpolacji wielomianem stopnia 0 dla 32 węzłów na przedziale [0;1]

Na wykresie (Rysunek 19) widać, że funkcja interpolująca przyjmuje tym bardziej zbliżone wartości do funkcji , im x jest bliższy zeru. Zaś przybliżenie jest mało dokładne, gdy x jest bliskie jeden, bo wtedy funkcja jest stroma.

Tabela 11: Wynik i błędy implementowanej funkcji dla f(x) = x^21 na przedziale [0;1] dla różnych e

A table of numbers and letters

Description automatically generated with medium confidence

A white background with a red line

Description automatically generated

Rysunek 20: Wykres dokładnej wartości całki z x^21 po [0;1] i jej przybliżenia w zależności od m

Tabela 12: Wynik i błędy implementowanej funkcji dla f(x) = x^21 na przedziale [0;1] dla pierwszych 10 wartości m

A table of numbers with black text

Description automatically generated

Z tabel (Tabela 11, Tabela 12) i wykresu (Rysunek 20) wynika, że całka jest dobrze przybliżona.

Podsumowanie wyników:

Jak widać na przedstawionych powyżej przykładach przybliżanie wartości działa najlepiej, czyli daje dokładne wyniki dla względnie małych m, dla funkcji, które w przedziale [a;b] są gładkie, nie mają gwałtownych wahań wartości i nie są strome, np. na [0;1] i oraz na [5;6], które miały błąd względny mniejszy niż już dla m = 128.

Zaś dla funkcji z gwałtownymi wahaniami wartości np. na [0;1] i , gdzie w zależności od tego w jakie punkty są węzłami interpolacji wynik może się znacznie różnić. Dlatego dopiero wysokie wartości m pozwolą otrzymać wynik z pożądaną dokładnością.s