Apraksts

Šajā laboratorijas darbā tika izpētīts funkcijas atvasinājuma algoritms, pēc kura tiek noteiktas atvasinātās funkcijas vērtības.

Formula un kods

Funkcijas atvasinājuma definīcjas formula ir:

$$y' = \frac{dy}{dx} = \lim_{\Delta x \to 0} \frac{f(x+\Delta) - f(x)}{\Delta x}$$

kurā tiek skatīts limits no argumenta izmaiņas un funkcijas vērtību izmaiņas attiecības. Taču šo procesu atvieglosim ar to, ka paņemsim ļoti mazu argumenta difrenciāli, iegūsim jaunu funkcijas vērtību un no tā iegūsim funkcijas vērtību diferenciāli, kuru izdalīsim ar argumenta diferenciāli.

Pirmā atvasinājuma kods un grafiks

```
printf("Cien.lietotāj, ievadiet a vērtību:\n");
scanf("%lf",&a);
printf("Cien.lietotāj, ievadiet b vērtību:\n");
scanf("%lf",&b);
printf("Cien.lietotāj, ievadiet dx vērtību:\n");
scanf("%lf",&delta_x);
```

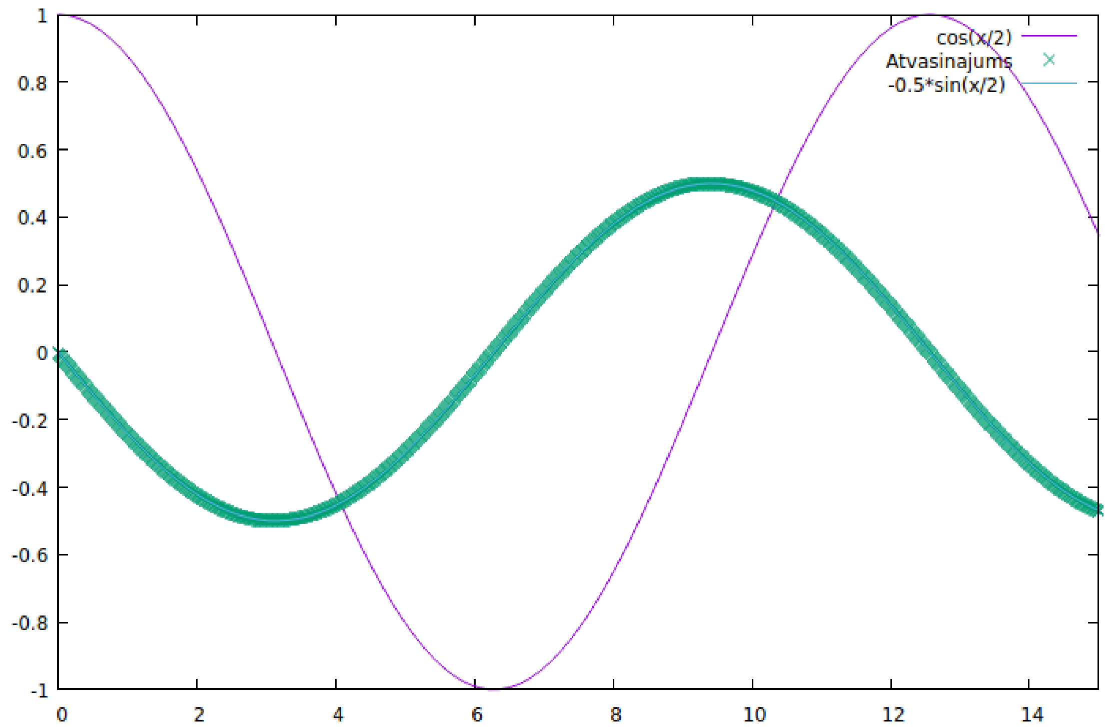
Ilustrācija 1. Vispirms tiek prasīts lietotājam, lai ievada vērtības.

```
printf("\tx\tcos'(x/2)\n");
while(x<b){
        y = cos(x/2);
        y1 = cos((x+delta_x)/2);
        y2 = (y1 - y)/delta_x;
        printf("%10.5f\t%10.5f\t\n",x,y2);
        x += delta_x;
}
```

Ilustrācija 2. Tālāk tiek noteiktas funkcijas vērtības pie argumentiem ar delta_x, kā rezultātā var iegūt funkcijas diferenciāli un 1.kārtas atvasinājumu. Cikla beigās delta_x tiek pieskaitīts argumentam, lai iegūtu vairākas atvasinātās funkcijas vērtības.

```
Cien.lietotāj, ievadiet a vērtību:
0
Cien.lietotāj, ievadiet b vērtību:
2
Cien.lietotāj, ievadiet dx vērtību:
0.01
        x       cos'(x/2)
   0.00000        -0.00125
   0.01000        -0.00375
   0.02000        -0.00625
   0.03000        -0.00875
   0.04000        -0.01125
   0.05000        -0.01375
```

Ilustrācija 3. Izvadītie rezultāti oriģinālfunkcijai un 1.kārtas atvasinātai funkcijai.



*Ilustrācija 4. 3 funkcijas grafiki: kosinuss no pusargumenta (oriģinālfunkcija), "atvasinājums" (iegūtie dati no definīcijas formulas), -0.5*sin(x/2) (funkcija, lai pārliecinātos, ka dati atbilst 1.kārtas atvasinājumam, datiem ar grafiku jāsakrīt).*

Atvasinātā funkcija, kura tika ievadīta kā funkcija, tika iegūta:

$$y' = \frac{dy}{dx}\cos\left(\frac{x}{2}\right) = -\sin\left(\frac{x}{2}\right) * \frac{dy}{dx}\left(\frac{1}{2}x\right) = -\frac{1}{2}\sin\left(\frac{x}{2}\right)$$

Rezultātā var pārliecināties, ka sastādītais algoritms ir precīzs atvasinātās funkcijas vērtību noteikšanai.

Otrā atvasinājuma kods un grafiks

```
y = cos(x/2);
y1 = cos((x+delta_x)/2);
y2 = (y1-y)/delta_x;
//printf("%10.5f\t%10.5f\t\n",x,y4);
x += delta_x;
y0 = cos(x/2);
y1 = cos((x+delta_x)/2);
y3 = (y1-y0)/delta_x;
y4 = (y3-y2)/delta_x;
printf("%10.5f\t%10.5f\t\n",x,y4);
```

Ilustrācija 5. Kods 2.atvasinājuma iegūšanai.

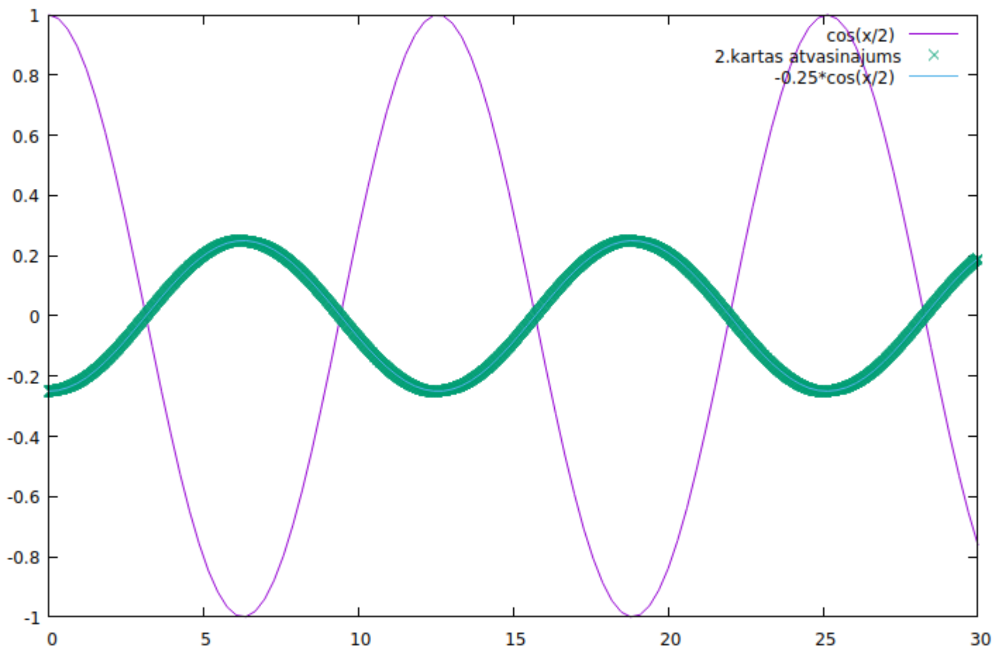
2.kārtas atvasinājums strādā sekojoši:

$$y'' = \frac{d(y')}{dx}$$

tāpēc, kad tiek iegūta pirmā 1.atvasinājuma vērtība, pēc tam pieskaita *dx* un atkal nosaka 1.atvasinājumu. Kad ir iegūti 2 1.atvasinājuma vērtības, tad nosaka diferenciāli un iegūst otro atvasinājumu. Šajā kodā y_2 ir pirmā 1.kārtas atvasinājuma vērtība, y_3 ir otrā 1.kārtas atvasinājuma vērtība un y_4 ir iegūtais 2.kārtas atvasinājums.

```
Cien.lietotāj, ievadiet a vērtību:
0
Cien.lietotāj, ievadiet b vērtību:
2
Cien.lietotāj, ievadiet dx vērtību:
0.01
        x       cos''(x/2)
   0.01000        -0.25000
   0.02000        -0.24999
   0.03000        -0.24997
   0.04000        -0.24995
   0.05000        -0.24992
   0.06000        -0.24989
```

Ilustrācija 6. Izvadītie dati oriģinālfunkcijai un 2.kārtas atvasinātai funkcijai.



Ilustrācija 7. 3 funkcijas grafiki: cos(x/2)(oriģinālfunkcija), dati, kuri tika iegūti no 2.kārtas algoritma un -0.25cos(x/2)(2.kārtas atvasinātā funkcija oriģinālfunkcijai, kura tika ievadīta pa taisno). Ja dati sakrīt ar 3.funkciju, tad algoritms strādā pareizi.

2.kārtas atvasinājums oriģinālfunkcijai ir:

$$y'' = \frac{dy}{dx}(-\frac{1}{2}\sin\left(\frac{x}{2}\right)) = -\frac{1}{2}\cos\left(\frac{x}{2}\right) * \frac{dy}{dx}\left(\frac{1}{2}x\right) = -\frac{1}{4}\cos\left(\frac{x}{2}\right)$$

Kā jau iepriekš minēju, ja algoritma iegūtie dati sakrīt ar to funkciju, kas tika aprēķināta un ievadīta, tad var secināt, ka algoritms strādā, kā ir nepieciešams.

Secinājumi

Izpētot un salīdzinot abu atvasinājumu algoritmus, var secināt, ka tie atbilst funkciju atvasinājumu noteikšanai ar lielu precizitāti. Konkrēti šajā gadījumā tika izmantota funkcija $f(x) = cos(x/2)$, taču kods strādā arī ļoti labi, ja tam pielāgo citu funkciju.