**Aplicação de Derivadas**

**Autor(a):** *Luiz Henrique Botega Beraldi.*

**Colaboradores(as):** *Prof. Dr. Robinson Samuel Vieira Hoto.*

**Contato:** *luizhenrique.beraldi@uel.br.*

***Resumo:*** *Neste relatório irei abordar sobre a Aplicação de Derivadas, irei introduzir os problemas diários para qual elas são usadas, explicando um pouco o que eles são exatamente e como funcionam. Também irei mostrar a modelagem matemática de cada problema, com as contas utilizadas, explicando para que elas servem e onde se situam no problema. A modelagem computacional também será mostrada, onde mostro como criei os códigos para cada problema e explico como que cada parte dele funciona, e por fim, irei concluir sobre tudo que foi apresentado durante o relatório, colocando minha opinião sobre o problema, além de citar minhas dificuldades e quem me ajudou a desenvolver algumas partes.*

**1.Introdução as aplicações**

Talvez a mais difundida aplicação das derivadas nas universidades seja na otimização de problemas. Onde utilizamos as derivadas para obter a maximização ou minimização de um determinado fenômeno.[1]

Os exemplos de otimização são a minimização de consumo de material, maximização do lucro em função das despesas, maximização da área em função do seu perímetro e otimização do tempo na produção industrial.

As derivadas também são aplicadas na física, como por exemplo temos a velocidade que é derivada do espaço em relação ao tempo e a aceleração que é a derivada da velocidade em relação ao tempo, assim temos a equação que descreve a posição de uma determinada partícula, conseguimos obter a equação da velocidade e da aceleração.

Elas podem ser aplicadas na biologia também na parte de estudo sobre crescimento populacional, O tamanho de uma certa população em um determinado instante é dado pelas taxas de natalidade e de mortalidade, as taxas relacionadas são duas ou mais quantidades variando simultaneamente entre si, logo são derivadas.

A administração também é uma das áreas que as derivadas podem ser usadas, pois em grande parte das vezes são coisas que envolvem a maximização de lucro ou a minimização de custos, como apresentado na otimização. Na linguagem mais especifica que é usada por economistas e administradores, o custo marginal, que nada mais é do que a variação do custo total de produção em função da quantidade de um produto produzido, essa variação é expressada através da derivada do custo total pela quantidade produzida.

As Derivadas possuem muitas aplicações várias áreas, como na medicina (como exemplo a concentração de uma substância no organismo), química (como exemplo na Lei de Boyle) entre outras.

Irei demonstrar com exemplos na modelagem matemática e computacional algumas partes das otimizações (minimização de consumo de material e maximização do lucro em função das despesas) e da parte da física (velocidade que é derivada do espaço em relação ao tempo e a aceleração que é a derivada da velocidade em relação ao tempo) já que são mais interativas.

Nas próximas partes do trabalho, irei mostrar as contas usadas para cada aplicação, mostrando como são feitas e explicando as contas, depois irei mostrar como faze-las em forma de código na modelagem computacional para que sejam feitas mais rapidamente já que é só colocar os números desejados no código e ele dará o resultado esperado que iria demorar para fazer a mão, e por fim, irei mostrar minhas dificuldades e o que eu achei desse artigo nas minhas considerações finais.

**2.Modelagem Matemática**

A derivada matemática é um conceito fundamental da análise matemática que descreve a taxa de variação instantânea de uma função em um ponto específico. Sua história remonta a alguns dos matemáticos mais influentes da história, como Isaac Newton e Gottfried Wilhelm Leibniz, que desenvolveram a teoria das derivadas no final do século XVII. A derivada tem uma ampla gama de aplicações em ciências, engenharia, economia e muitos outros campos.

A derivada é uma ferramenta poderosa para entender e analisar funções matemáticas. Ela fornece informações sobre a inclinação ou taxa de variação da função em um ponto específico.

Em resumo, a derivada é uma ferramenta essencial na matemática e nas ciências aplicadas. Ela permite uma compreensão profunda do comportamento das funções e desempenha um papel fundamental na resolução de problemas em diversas áreas. Seu desenvolvimento ao longo da história da matemática foi crucial para o avanço do conhecimento e da tecnologia.

**Origens da Derivada:**

1. **Gregorius Amandus:** Embora a noção de derivada não tenha sido formalizada até o final do século XVII, o matemático holandês Gregorius Amandus (Gregoire de Saint-Vincent) fez contribuições iniciais relacionadas a tangentes e taxas de variação no século XVII.
2. **Pierre de Fermat:** O matemático francês Pierre de Fermat também fez avanços significativos no cálculo das taxas de variação e tangentes no início do século XVII. Ele é frequentemente creditado com a descoberta da tangente a uma curva em um ponto específico.
3. **Isaac Newton e Gottfried Wilhelm Leibniz:** No final do século XVII, Isaac Newton na Inglaterra e Gottfried Wilhelm Leibniz na Alemanha desenvolveram formalmente o cálculo, incluindo o conceito de derivada. Newton usou notação de diferenças finitas, enquanto Leibniz introduziu a notação de derivadas como a conhecemos hoje. Suas contribuições foram independentes, mas essencialmente equivalentes.

Irei fazer um simples cálculo de derivada mostrando como ela funciona.

Vamos considerar um exemplo simples e calcular a derivada da função f(x) = x^2 no ponto x = 2.

**Calcular a Derivada da Função:**

A função é uma função quadrática. Para calcular sua derivada, usamos a regra de potência. A regra de potência estabelece que a derivada de é . Neste caso, .

A derivada de f(x) é .

**Encontrar a Inclinação da Tangente no Ponto :**

Agora que temos a derivada f'(x) = 2x, podemos calcular a inclinação da tangente à curva no ponto .

Substituímos x por 2 na derivada:

A inclinação da tangente à curva de no ponto é igual a 4.

Isso significa que, no ponto , a curva é inclinada para cima com uma inclinação de 4. Você pode pensar nisso como a taxa de variação instantânea da função nesse ponto.

**Visualizar a Tangente:**

Para visualizar a tangente, você pode representar a reta tangente no ponto (2, 4) no gráfico da função. A equação da reta tangente será da forma , onde m é a inclinação da tangente (que calculamos como 4) e (2, 4) é o ponto de tangência.

A equação da reta tangente é, portanto:

Agora, podemos usar o ponto (2, 4) para encontrar o valor de b:

Subtraindo 8 de ambos os lados:

Portanto, a equação da tangente à curva de no ponto é:

Você pode agora plotar essa reta no gráfico da função para visualizar como a tangente se relaciona com a curva no ponto .

Para facilitar o cálculo de uma derivada, podemos usar as regras de derivação, para podermos calcular outras

funções e não ter que calcular toda vez pelo limite.

Irei mostrar agora as propriedades das derivadas, ou seja, as regras de derivação. Considere e duas funções

que sejam deriváveis e uma constante e que

* Constante vezes Função

* Derivada da Soma

* Derivada da Diferença

* Regra do Produto

* Regra do Quociente

* Regra da Cadeia

Ainda existem algumas outras propriedades, que são usadas para calcular potências, logaritmos, número de Euler e funções trigonométricas.

**3.Exemplos**

Aqui irei mostrar e explicar a modelagem matemática das aplicações, com as contas que foram usadas para ele ser feito e explicando para que serve o que está sendo apresentado de acordo com a aplicação que estamos resolvendo.

Vamos começar com as aplicações de otimização, onde temos:

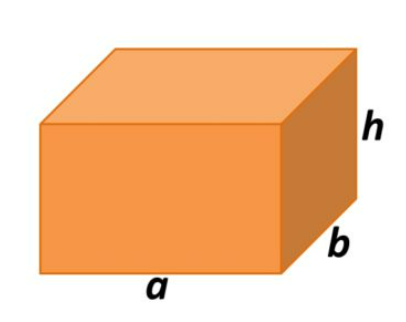
* Minimização do consumo de material;
* Maximização do lucro em função das despesas;

**3.1Minimização do consumo de material**

Dentro das aplicações da derivada nos deparamos com os Problemas de Máximos e Mínimos, que consiste em encontrar o maior e o menor valor da função em um determinado intervalo.[2]

Vou propor um problema simples enfrentado por uma empresa e iremos decorrer a explicação sobre ele.

“Uma empresa de embalagem recebeu um pedido de caixas de papelão, onde o solicitante exigiu apenas que as caixas tivessem 15 litros de capacidade e uma altura de 20 centímetros. Quais são as dimensões das caixas para obter o menor custo com o papelão?”, a Imagem 1 mostra o paralelepípedo do qual estou falando.



*Imagem 1: Demonstra o paralelepípedo que estou descrevendo*

Irei começar fazendo uma legenda para identificarmos o que é cada coisa nas contas, para facilitar a visualização do que está sendo feito e assim poderemos ter melhor entendimento da função de cada conta para a resolução do problema:

A altura da caixa será representada por h.

O volume da caixa será representado por V.

O comprimento será a.

E a largura será b.

Pois bem, vamos começar as contas então:

h = 20cm e V = 15l.

Primeiramente, devemos usar sempre as mesmas unidades de medida quando estamos trabalhando com algo, assim, devemos transformar litros em centímetros cúbicos da seguinte maneira:

1l = , assim V = .

Agora, devemos encontrar as dependências entre as variáveis do problema para obter o volume e a área da seguinte maneira:

Volume:  V =

Area:

Pode-se observar que multiplicamos toda a função por dois, já que os lados de um paralelepípedo reto são iguais.

Pois bem, agora devemos achar a função que relacione as variáveis a serem minimizadas, pois o que queremos, é obter o menor consumo de papelão.

Pela equação usada no volume encontramos:

onde 15.000 =

e substituindo na equação da área as informações obtemos:

Derivando em relação a variável b, fica-se com:

É bom lembrar que os pontos de máximos e mínimos são os pontos nos quais o resultado da derivada é nula, ou seja, . Assim, manipulando e substituindo a expressão obtém-se:

Entretanto, o valor que possui sentido físico no problema que estamos trabalhando é uma medida positiva, assim . Por fim, devemos encontrar a outra dimensão:

Então, a caixa de papelão que foi citada no enunciado do problema, tem que ter um fundo quadrado de lado com 27,386cm, para assim, obtermos o menor consumo de papelão possível na sua fabricação.

Esse problema é bem simples e bobo, mas mostra exatamente como funciona a minimização do consumo de material, no exemplo é só um paralelepípedo e o consumo é de papelão, mas imagine uma empresa, com inúmeros funcionários, maquinas, materiais, o quanto menos for gasto de seus materiais, melhor, aproveitando tudo o que pode, minimizando também os gastos, um paralelepípedo mal embrulhado não faz tanta diferença, mas imagine 100, 1000, 10.000, etc, o gasto seria enorme.

**3.2Maximização do lucro em função das despesas**

Nestes problemas necessitamos encontrar os pontos onde a função dada assume seu maior valor (máximo) ou menor valor (mínimo) em um determinado intervalo.[3]

Novamente irei propor um problema para entendermos melhor, agora, um problema enfrentado por um fazendeiro e suas cabeças de boi.

“Um fazendeiro tem 200 bois, cada um pesando 300 Kg. Até agora ele gastou R$380.000,00 para criar os bois e continuará gastando R$ 2,00 por dia para manter cada boi. Os bois aumentam de peso a uma razão de 1,5 Kg por dia. Seu preço de venda, hoje é de R$ 18,00 o quilo, mas o preço cai 5 centavos por dia. Quantos dias deveria o fazendeiro aguardar para maximizar seu lucro?”

Irei começar fazendo uma legenda para identificarmos o que é cada coisa nas contas, para facilitar a visualização do que está sendo feito e assim poderemos ter melhor entendimento da função de cada conta para a resolução do problema:

Os bois serão representados por b.

O peso de cada boi será p.

Pois bem, vamos começar as contas então:

b = 200.

p = 300Kg.

Custo até o momento: 380.000 reais.

Lucro = (peso total dos bois X preço ao Kg de cada boi) – custo total da criação

Vamos obter o resultado da função mostrada acima separadamente, em função de dias, pois o problema citado nos pede a quantidade de dias para otimizar o lucro.

* **Peso total dos bois:**

No problema citado, o enunciado nos diz que os 200 bois da fazenda pesam, cada um, 300Kg e que engordam 1,5 kg ao dia, assim temos que o peso total dos bois é obtido na seguinte função:

Peso total dos bois *= .*

* **Preço*ao Kg de cada boi:***

O preço do Kg também está variando com o passar dos dias na relação de que o preço do quilo, que é R$18,00 cai R$0,05 de valor a cada dia que se passa:

Preço ao Kg de cada boi = .

* **Custo total da criação dos bois:**

Até o agora o fazendeiro do problema teve um custo de R$ 380.000, mas, a cada dia a mais que os bois ficam na fazenda, eles geram mais custos da seguinte maneira:

Custo total da criação dos bois = .

Substituindo a equação geral obtém-se a seguinte equação:

Lucro =

Realizando as operações da função acima ficaremos com o seguinte resultado de lucro:

Lucro = .

Depois de tudo, o que realmente queremos é o lucro máximo, então o que devemos fazer é achar o ponto exato onde a função de lucro alcança o maior valor possível. Para podermos achar isso, aplicaremos a derivada na função e igualaremos a mesma a zero, já que nos pontos onde a derivada é nula, ou seja, igual a zero, temos os pontos de máximo e de mínimo:

Então, o fazendeiro do problema deve o aguardar 67 dias para maximizar seu lucro de sua fazenda.

Ambos os dois exemplos, usaram máximo e mínimo, o que é excelente para o que fizemos nos exercícios, a otimização, para não perder material e maximizar o lucro da empresa.

Agora irei demonstrar exemplos da aplicação de derivadas na física, onde temos dois exemplos clássicos, que são:

* Velocidade que é derivada do espaço em relação ao tempo;
* Aceleração que é a derivada da velocidade em relação ao tempo ;

**3.3Velocidade que é derivada do espaço em relação ao tempo**

Dentro das aplicações de derivadas, temos a derivação do espaço em relação ao tempo que é a velocidade, essa

derivada é muito utilizada especialmente na física, mas também no nosso dia-a-dia.

A fórmula para a velocidade instantânea de um objeto em movimento é definida como a derivada da posição

em relação ao tempo. Matematicamente, pode ser expressa da seguinte forma:

“”

Onde:

* *v*(*t*) é a velocidade instantânea no instante de tempo *t*;
* representa a derivada em relação ao tempo;
* é a posição do objeto no instante de tempo *t*.

Em palavras, a velocidade instantânea é a taxa de mudança da posição em relação ao tempo e nos diz a que taxa

a posição do objeto está mudando em um determinado momento no tempo. A unidade típica de velocidade no Sistema Internacional (SI) é metros por segundo (m/s).

**3.4Aceleração que é a derivada da velocidade em relação ao tempo**

Dentro das aplicações de derivadas, temos a derivação da velocidade em relação ao tempo que é a aceleração,

essa derivada é muito utilizada especialmente na física, mas também no nosso dia-a-dia.

A fórmula para a aceleração (*a*) é a derivada da velocidade (*v*) em relação ao tempo (*t*). Matematicamente, pode

ser expressa da seguinte forma:

“”

Onde:

* é a aceleração no instante de tempo t;
* representa a derivada em relação ao tempo;
* é a velocidade do objeto no instante de tempo t.

Em palavras, a aceleração é a taxa de mudança da velocidade em relação ao tempo e nos diz a que taxa a velocidade do objeto está mudando em um determinado momento no tempo. A unidade típica de aceleração no Sistema Internacional (SI) é metros por segundo ao quadrado (m/s²).

Agora, iremos fazer a modelagem computacional das contas mostradas acima, sendo muito mais fácil apenas colocar os valores desejados no código e obter o resultado do que fazer todas as contas à mão, podendo facilmente errar nas contas e prejudicar e muito a empresa.

**4.Modelagem Computacional**

Aqui irei explicar em termos computacionais da linguagem C as contas que mostrei na modelagem matemática, já que aqui a pessoa só coloca os valores e obtém o resultado esperado, sem medo de ter contas erradas, já que os códigos foram completamente revisados.

**4.1Aplicações de otimização**

**4.1.1Minimização do consumo de material**

Irei começar pela “Minimização do consumo de material”, o código começa adicionando a biblioteca “stdio.h” e “math.h” para poder funcionar corretamente.

Já na funão “main”, crio as variáveis “h” e “V” do tipo “double”, que significam respectivamente a altura da caixa em centímetros e o volume da caixa em centímetros cúbicos.

Faço dois “printf” pedindo que o usuário digite os valores de “h” e “V” e guardando os valores digitados nas respectivas variáveis.

Então crio as variáveis “a” (comprimento), “b” (largura), “A” (área) e “min\_A” (área mínima) que é iniciado com um valor extremamente grande para evitar erros no código, eu o iniciei com “1000000000000000000,0”, todos são do tipo “double” também.

A seguir, uso um loop “for” em que “b” começa em 1.0 e acrescenta de 0.001 a 0.001, o loop irá acontecer até que “b” seja maior ou igual a 100.

Dentro do loop “for”, faço “a” (comprimento) receber a seguinte conta:

“a = V / (h \* b)”

Que serve para encontrar o valor de “a” a partir do volume e altura.

Ainda dentro do loop “for”, faço a variável “A” (área) receber a seguinte conta:

“A = 2 \* (a \* b + a \* h + b \* h)”

Que serve para calcular a área.

Embaixo disso, temos um comparador “if” que verifica se “A” é menor que “min\_A”, caso seja, “min\_A” recebe o que está em “A”, se não, é usado um “break” para que o loop seja encerrado quando a área começar a aumentar, já que o código busca a área mínima.

Por fim, é mostrado na tela, usando dois “printf”, os resultados de “a” que contém o comprimento mínimo necessário e “b” que contém a largura mínima necessária para fazer a embalagem do paralelepípedo com a menor utilização de papelão possível.

O código em questão:

1. #include <stdio.h>
2. #include <math.h>
3. int main() {
4. // Dados do problema
5. double h; // Altura da caixa em centímetros
6. double V; // Volume da caixa em centímetros cúbicos
7. printf("\nDigite a altura:");
8. scanf("%lf",&h);
9. printf("\nDigite o volume:");
10. scanf("%lf",&V);
11. // Encontrar a área mínima
12. double a, b, A, min\_A = 1000000000000000000.0; // Inicializar min\_A com um valor grande
13. for (b = 1.0; b <= 100.0; b += 0.001) {
14. a = V / (h \* b); // Encontrar o valor de 'a' a partir do volume e altura
15. A = 2 \* (a \* b + a \* h + b \* h); // Calcular a área
16. if (A < min\_A) {

min\_A = A; // Atualizar a área mínima

1. } else {

break; // Parar o loop quando a área começar a aumentar

1. }
2. }
3. // Resultados
4. printf("\nA caixa de papelao deve ter as seguintes dimensoes:\n");
5. printf("\nComprimento (a): %.2lf cm\n", a);
6. printf("\nLargura (b): %.2lf cm\n", b);
7. return 0;
8. }

**4.1.2Maximização do lucro em função das despesas**

Agora irei explicar a “Maximização do lucro em função das despesas”, o código novamente começa adicionando a biblioteca “stdio.h” para o código funcionar corretamente, já que sem essa biblioteca o código não iria compilar.

Já na função “main”, crio as variáveis “b” (número de bois) e “d” (dias) que são do tipo “int” já que não colocaremos números com ponto flutuante nelas, as variáveis em que iremos utilizar isso eu irei apresentar agora, as variáveis “p” (peso dos bois), “custo\_total” e “lucro\_maximo” que já pelo nome são auto explicativas, essas três variáveis são do tipo “float”.

Na sequência, peço para o usuário digitar o número de bois, o peso deles (no caso seria uma média do peso de todos os bois, já que só se pode digitar um valor aqui) e o custo total, todos esses valores são guardados respectivamente nas variáveis mostradas acima.

Então, crio um loop “for” que não tem uma condição de parada, não é algo tão utilizado mas ainda sim tem suas características boas e nesse código foi bem utilizado, faço o “d” começar em 0 e o loop “anda” adicionando 1 à variável “d”.

Dentro do loop, crio a variável “preco\_por\_quilo” do tipo “float” que calcula o preço por quilo, que diminui cinco centavos por dia, a conta é feita da seguinte forma:

“preco\_por\_quilo = 18,0 – 0,05 \* d”

Após isso, crio outra variável do tipo “float”, a variável “peso\_total” que calcula o peso total dos bois, levando em consideração o ganho diário de peso, que é calculado da seguinte forma no código:

“peso\_total = b \* p + b \* 1,5 \* d”

Novamente, crio outra variável do tipo “float”, chamada “custo\_diario” que calcula o custo diário, assumindo um custo fixo de R$2,00 por boi por dia, e a conta é feita da seguinte forma:

“custo\_diario = 2,0 \* b \* d”

E por fim, crio outra variável do tipo “float”, chamada “lucro” que calcula o lucro para o dia atual, sendo feita da seguinte maneira:

“lucro = (peso\_total \* preco\_por\_quilo) – (custo\_total + custo\_diario)”

Então crio um comparador “if” que verifica se “lucro” é menor que “lucro\_maximo”, se o lucro for menor que o lucro máximo encontrado até agora o loop é encerrado com um “break”, e ainda dentro do loop “for” faço a variável “lucro\_maximo” receber “lucro” caso o lucro atual for maior, então o lucro máximo passa a ser o lucro atual.

Por fim, quando o loop é encerrado, é imprimido na tela com um “printf” o tanto de dias que o fazendeiro deve aguardar para o obter o lucro máximo, que é obtido pela variável “d” ao final do loop.

O código em questão:

1. #include <stdio.h>
2. int main() {
3. int b, d;
4. double p, custo\_total, lucro\_maximo = 0.0;
5. // Solicita ao usuário que insira o número de bois
6. printf("\nDigite o numero de bois:");
7. scanf("%d", &b);
8. // Solicita ao usuário que insira o peso de cada boi
9. printf("\nDigite o peso de cada boi:");
10. scanf("%lf", &p);
11. // Solicita ao usuário que insira o custo total
12. printf("\nDigite o custo total:");
13. scanf("%lf", &custo\_total);
14. for (d = 0; ; d++) {
15. // Calcula o preço por quilo, que diminui 5 centavos por dia
16. double preco\_por\_quilo = 18.0 - 0.05 \* d;
17. // Calcula o peso total dos bois, levando em consideração o ganho diário de peso
18. double peso\_total = b \* p + b \* 1.5 \* d;
19. // Calcula o custo diário, assumindo um custo fixo de R$ 2,00 por boi por dia
20. double custo\_diario = 2.0 \* b \* d;
21. // Calcula o lucro para o dia atual
22. double lucro = (peso\_total \* preco\_por\_quilo) - (custo\_total + custo\_diario);
23. // Se o lucro for menor que o lucro máximo encontrado até agora, encerra o loop
24. if (lucro < lucro\_maximo) {

break;

1. }
2. // Atualiza o lucro máximo se o lucro atual for maior
3. lucro\_maximo = lucro;
4. }
5. // Imprime o número de dias que o fazendeiro deve aguardar para maximizar o lucro
6. printf("\nO fazendeiro deve aguardar %d dias para maximizar seu lucro.\n", d - 1);
7. return 0;
8. }

**4.2Aplicações na Física**

**4.2.1Velocidade que é derivada do espaço em relação ao tempo**

Irei começar pela Velocidade que é derivada do espaço em relação ao tempo, o código começa adicionando as

bibliotecas“stdio.h” e “math.h” para que o código funcione corretamente.

Após isso, crio as variáveis “x0” (posição inicial), “x1” (posição final), “t0” (tempo inicial) e “t1” (tempo final),

todas elas são do tipo “float”.

Usando alguns “printf”, eu solicito ao usuário que digite a posição inicial (x0) e o tempo inicial (t0), após isso, peço novamente para ele digitar, só que dessa vez a posição final (x1) e o tempo final (t1), com todos os valores sendo guardados dentro de suas respectivas variáveis.

Então, crio uma variável do tipo “double”, chamada “velocidade”, que calcula a velocidade instantânea usando a formula v(t) = (x1 – x0) / (t1 – t0), que fica da seguinte forma no código:

“velocidade = (x1 – x0) / (t1 – t0)”

Após isso, eu imprimo na tela a velocidade instantânea em metros por segundo que foi obtido na variável “velocidade” e encerro o programa.

O código em questão:

1. #include <stdio.h>
2. #include <math.h>
3. int main() {
4. double x0, x1, t0, t1;
5. // Solicite ao usuário que insira a posição inicial (x0) e o tempo inicial (t0)
6. printf("\nInsira a posicao inicial (x0): ");
7. scanf("%lf", &x0);
8. printf("\nInsira o tempo inicial (t0): ");
9. scanf("%lf", &t0);
10. // Solicite ao usuário que insira a posição final (x1) e o tempo final (t1)
11. printf("\nInsira a posicao final (x1): ");
12. scanf("%lf", &x1);
13. printf("\nInsira o tempo final (t1): ");
14. scanf("%lf", &t1);
15. // Calcule a velocidade instantânea usando a fórmula v(t) = (x1 - x0) / (t1 - t0)
16. double velocidade = (x1 - x0) / (t1 - t0);
17. // Exiba o resultado
18. printf("\nA velocidade instantanea eh %.2lf metros por segundo (m/s).\n", velocidade);
19. return 0;
20. }

**4.2.2Aceleração que é a derivada da velocidade em relação ao tempo**

Agora irei mostrar a aceleração que é a derivada da velocidade em relação ao tempo, o código começa

adicionando as bibliotecas“stdio.h” e “math.h” para que o código funcione corretamente.

Após isso, crio as variáveis “v0” (velocidade inicial), “v1” (velocidade final), “t0” (tempo inicial) e “t1” (tempo

final), todas elas são do tipo “float”.

Usando alguns “printf”, eu solicito ao usuário que digite a velocidade inicial (v0) e o tempo inicial (t0), após isso, peço novamente para ele digitar, só que dessa vez a velocidade final (v1) e o tempo final (t1), com todos os valores sendo guardados dentro de suas respectivas variáveis.

Então, crio uma variável do tipo “double”, chamada “aceleracao”, que calcula a velocidade instantânea usando a formula v(t) = (v1 – x0) / (v1 – t0), que fica da seguinte forma no código:

“aceleracao = (x1 – x0) / (t1 – t0)”

Após isso, eu imprimo na tela a aceleração em metros por segundo que foi obtido na variável “velocidade” e encerro o programa.

O código em questão:

1. #include <stdio.h>
2. #include <math.h>
3. int main() {
4. double v0, v1, t0, t1;
5. // Solicite ao usuário que insira a velocidade inicial (v0) e o tempo inicial (t0)
6. printf("Insira a velocidade inicial (v0) em m/s: ");
7. scanf("%lf", &v0);
8. printf("Insira o tempo inicial (t0) em segundos: ");
9. scanf("%lf", &t0);
10. // Solicite ao usuário que insira a velocidade final (v1) e o tempo final (t1)
11. printf("Insira a velocidade final (v1) em m/s: ");
12. scanf("%lf", &v1);
13. printf("Insira o tempo final (t1) em segundos: ");
14. scanf("%lf", &t1);
15. // Calcule a aceleração usando a fórmula a(t) = (v1 - v0) / (t1 - t0)
16. double aceleracao = (v1 - v0) / (t1 - t0);
17. // Exiba o resultado
18. printf("A aceleração é %.2lf metros por segundo ao quadrado (m/s²).\n", aceleracao);
19. return 0;
20. }

**Conclusões**

Pois bem, não tive tantas dificuldades assim com esse artigo, mas o que obtive foi um certo conhecimento sobre derivadas e onde são aplicadas.

Eu sabia de algumas de suas aplicações, mas não sabiam que eram tão diretas no nosso dia-a-dia, na parte de otimização mesmo, mostra como todas as empresas abordam as derivadas de alguma forma, mesmo sem saber, para poder utilizar ao máximo seu maquinário e economizar dinheiro ao saber exatamente o quanto é o máximo que pode gastar ao fabricar algum material.

As suas aplicações na física eu já tinha ouvido falar sobre, mas nunca pesquisei tão a fundo como agora, e elas são extremamente interessantes e simples de entender, sendo utilizadas nos radares e medidores de velocidade.

Não discorri muito no artigo, mas pesquisei também sobre a derivada sendo utilizada na administração e na biologia, além de outros casos, como a Lei de Boyle, vale a pena a curiosidade sobre os mesmos, sendo totalmente interessantes.

As mínimas dificuldades que tive foi em explicar as contas de cada derivada, já que tive que entender todas elas antes e de alguma forma explicar para que todos que estão lendo este artigo possam entender de forma simples o que foi escrito, outra dificuldade foi para “traduzir” as contas em códigos em C, demorei um tempo para conseguir entender como faria cada uma mas não foi uma grande dificuldade, olhando agora, percebo o quão simples são os códigos, também expliquei eles de uma maneira simples para que até aqueles que não tem muito contato com programação ou com a linguagem C possam replicar os códigos sem muita dificuldade caso queiram experimentar ou até mesmo testar para saber se tudo o que disse era realmente verdade.

O que também posso dizer é que este tipo trabalho é realmente ótimo para nosso futuro, já que iremos fazer inúmeros artigos dos mais diversos assuntos, por mais que este artigo seja simples e sem muita linguagem formal, ele serve muito bem como treino para nos prepararmos para o que vem nos próximos anos.

**Bibliografia**

[1] Derivadas possuem diversas aplicações. In: Dicas de Cálculo. Disponível em: <https://www.dicasdecalculo.com.br/possuem-diversas-aplicacoes/>. Acesso em: 03 de outubro de 2023.

[2] Problemas de Máximos e Mínimos – Exercícios resolvidos. Disponível em: <https://www.dicasdecalculo.com.br/problemas-de-maximos-e-minimos-exercicios-resolvidos/>. Acesso em: 03 de outubro de 2023.

[3] Problemas resolvidos de máximos e mínimos – Exercício resolvido. Disponível em: <https://www.dicasdecalculo.com.br/problemas-resolvidos-de-maximos-e-minimos/>. Acesso em 03 de outubro de 2023.