

ATIVIDADE 3

1) Quem foi o idealizador do cálculo lambda e quando ele foi proposto (aproximadamente)?

O cálculo lambda foi desenvolvido pelo matemático Alonzo Church em 1930.

2) Qual a relação entre cálculo lambda e máquina de Turing?

O cálculo lambda e a máquina de Turing são as duas principais abstrações usadas para estudar a computabilidade e a teoria da computação. Qualquer problema que possa ser resolvido por uma máquina de Turing também pode ser resolvido por meio do cálculo lambda, e vice-versa.

3) O que é o cálculo lambda? Em que ele foi útil na computação?

O cálculo lambda é um sistema formal de notação matemática para representar funções e expressões matemáticas de forma abstrata, sem depender de uma linguagem de programação específica.

4) O que são variáveis livres (independentes)? E variáveis vinculadas (dependentes)? Cite exemplos de variáveis livres e vinculadas em uma expressão lambda

- Variáveis livres (ou independentes) não são declaradas como parâmetros em uma abstração lambda, mas que aparecem em sua expressão. $(\lambda x. x + y)$ - o "x".
- Variáveis vinculadas (ou dependentes) são aquelas que são declaradas como parâmetros em uma abstração lambda. $(\lambda x. x + y)$ - o "y".

5) O que significa currying em cálculo lambda? Exemplifique.

Consiste em transformar uma função que recebe múltiplos argumentos em uma sequência de funções que recebem apenas um argumento

6) O que significa uma expressão ser um "combinador", em cálculo lambda? Cite um exemplo de expressão que é um combinador

Todas as variáveis em sua expressão são vinculadas por abstrações lambda. $Ex = \lambda f. \lambda x. f\ x$

7) O que é aplicação e o que é abstração numa expressão lambda? Exemplifique

- A aplicação é a operação de aplicar uma função lambda a um argumento, produzindo um resultado.
- A abstração, por outro lado, é a operação de criar uma função lambda a partir de uma expressão lambda, vinculando uma variável em uma expressão lambda para criar uma nova função.

8) O que significa dizer que duas expressões lambda são α -equivalentes? Exemplifique.

Se duas expressões lambda têm a mesma estrutura e diferem apenas pelos nomes de suas variáveis vinculadas, elas são consideradas α -equivalentes. $(\lambda x. \lambda y. x + y == \lambda z. \lambda y. z + y)$

9) O que é a operação de α -conversão (α -renomeação)? Exemplifique.

A α -conversão é uma operação que preserva o significado da expressão lambda, mas altera apenas os nomes das variáveis vinculadas.

10) O que é Redução-Beta (β -Redução)?

Ela consiste em aplicar uma expressão lambda a um argumento, substituindo todas as ocorrências de sua variável vinculada pelo argumento.

11) O que significa dizer que uma expressão lambda está em sua "forma normal"? Exemplifique com uma expressão lambda em sua forma normal.

Uma expressão lambda está em sua forma normal quando não é possível realizar mais nenhuma redução-beta nessa expressão. Exemplo de expressão lambda em sua forma normal é a constante $\lambda x. 42$, que representa uma função que sempre retorna o valor 42.

12) Pesquise sobre o Combinador Y. O que é e o que ele faz? Descreva um pouco seu funcionamento.

O Combinador Y é uma expressão lambda que permite a recursão em cálculo lambda. Ele retorna uma nova expressão lambda que tem a mesma forma, mas com uma referência recursiva adicionada.

1) Crie uma expressão Lambda que receba dois valores inteiros (x e y) e retorne o resultado da soma dos dois valores. Leia os valores x e y do teclado.

```
soma = lambda x, y: x + y
x = int(input())
y = int(input())
print(soma(x, y))
```

2) Crie uma expressão Lambda que receba três notas de um aluno (a, b, c), calcule a média e retorne se o aluno foi aprovado ou reprovado. Para um aluno ser aprovado, ele deve possuir nota igual ou superior a 6. Leia as notas dos alunos do teclado.

```
aprovado = lambda a, b, c: "Aprovado" if ((a + b + c) / 3) >= 6 else "Reprovado"
```

```
a = float(input("Digite a primeira nota: "))
b = float(input("Digite a segunda nota: "))
c = float(input("Digite a terceira nota: "))
```

```
print(aprovado(a, b, c))
```