Revisão de POO - NOM2 Encapsulamento-Const unção definida fora do escopo da Extrair do Classes Métodos que não alteram o estado do mundo apenas asse para acessar Receita do seu objeto devem ser definidos como const o que é s membros públicos e privados relevante ao 'Uma classe se resume aos seus atributos no baixo dessa classe sistema nível" Objetos que não precisam/podem ser quebra de modificados, e, portanto, devem ser encapsulamento) Instância da classe É o tipo/sabor do bolo Objeto definidos como objetos const Destrutor Construtor Características que definem as Atributos Os igredientes do seu bolo propriedades de um objeto Mesmo nome da classe, sem retorno, pode ser sobrecarregado Função-membro especial utilizada para desalocar a memória usada para criação de determinados objetos. Função que altera ou utiliza DESCRIPTION OF THE PERSON OF T As ações para fazer o seu bolo Função membro especial que é invocada na definição do Métodos atributos para executar ações objeto para inicializá-lo. Essa função é chamada imediatamente antes que objeto Interface fica no seja destruído justamente para limpar a memória desse. Valores default: São valores que você define para um ou .hpp (ou .h) objeto antes que ele seja permanentemente excluído mais parâmetros de uma função, de modo que, caso o Implementação suário não forneça um valor ao chamar a função, o valor fica no .cpp C++ 20 em POO Ponteiro This padrão será utilizado (feito na .h) Métodos de acesso Métodos utilitários Argumento implicito presente em cada Métodos Getters e Interface fodificadores de Acess função-membro não status da classe Setters **Modularidade métodos public métodos private e Organização* São os métodos públicos da classe, o que ela oferece ao mundo. Setters: Modificam Recebem algo Contém o endereço do objeto a partir do Retornam nada valores de atributo qual foi invocado o método. **Flexibilidade (Reuso** Sintaxe de Inicialização Implementação Getters: Conseguem Recebem nada istinguir o parâmetro do atributo quando **Facilidade na valores em atributos Retornam algo Manutenção** estes possuirem o mesmo nome A sintaxe de inicialização de membros é colocada logo Como a classe realiza as operações e funcionalidades após o nome do construtor, pode ser usada sempre. definidas pela interface new and delete UML (Unified Modeling Language) std::string name, int toleto) : none(name), intele(idade) { Associação Permitem alocar e desalocar memória dinamicamente (em tempo de execução) Oto Eficiência: Inicializa os membros diretamente, sem Quando dois objetos se precisar de atribuições no corpo do construtor. relacionam de alguma form Login: String Assigns new[] delete[] -> Mesma coisa só que para Password: String Necessário para const e reference: Esses membros só arrays de objetos podem ser inicializados dessa forma, já que não podem Controls Composição Agregação ser atribuídos após a criação do objeto. Associação entre Project Report Employee classes: Membros Statics ID: Number Time: Number D: Number É um tipo de relação Retornar referência p/ membro private TosksiD: Number[] Login: String Date: Date Uma Classe tem objetos de que indica que uma Possword String Access: Number[] Took: String outra Classe como membr lasse está conectada a Project: String Name: String Propriedade da Classe compartilhada por outra de alguma todos os objetos dessa Classe Retornar uma referência para membros privados pode forma, permitindo que A parte e o todo são ser perigoso porque permite que o código externo Na composição, a parte faz ndependentes. O todo ainda objetos de uma classe modifique diretamente os dados da classe, quebrando parte do todo e o todo não existe mesmo sem a parte, e nterajam com objetos o encapsulamento e violando a integridade da classe. (sinal de menos): atributos/métodos privados a parte ainda existe sem o existe sem a parte. Sempre existe independente de objetos da outra. + (sinal de mais): atributos/métodos públicos