**Questões de Algoritmo e programação 2**

(Protótipo e Interface)

(Eduardo Dias das Neves)

Questão 1: Na linha 21 as inicias do aluno continua a mesma (sem alterações da função), entretanto na linha 23 é aplicado a função, deixando as iniciais maiúsculas.

A linha 23 se for substituída por “ cout << “ nome” << endl; ” no lugar destacar a mudança que a função realiza, vai mostrar o nome do aluno.

—-----------------------------------------------------------------------------------------------------------

Questão 2: Consegue descobrir se uma função tem retorno quando ela é chamada dentro do código após o “ int main () ” e principalmente no “ cout ” Parâmetro o que está dentro do Parênteses “(x)” .

—-----------------------------------------------------------------------------------------------------------

Questão 3: As funções listadas na imagem são funções padrão da biblioteca C utilizadas para gerar números aleatórios e converter strings em inteiros. Exemplos em detalhes sobre ela:

—-----------------------------------------------------------------------------------------------------------

Exemplo 1 → . Int rand(void): Esta função retorna um número pseudo aleatório inteiro entre 0 e “RAND\_MAX” (um valor constante definido na biblioteca padrão). A função não aceita parâmetros, por isso a declaração utiliza “void”.

Exemplo 2 →. void srand(unsigned int seed): Esta função inicializa o gerador de números pseudo aleatórios com um valor específico chamado “seed”. Usar uma semente diferente permite a geração de diferentes sequências de números aleatórios. A função não retorna nenhum valor (`void`).

Exemplo 3 → int atoi(const char \*str): Esta função converte uma string (const char) que representa um número inteiro em um valor “int”. Se a string começar com caracteres que não representam um número, o comportamento da função não é definido.

Essas funções são comuns em programação em C e são amplamente utilizadas para manipulação básica de números e strings.

—-----------------------------------------------------------------------------------------------------------