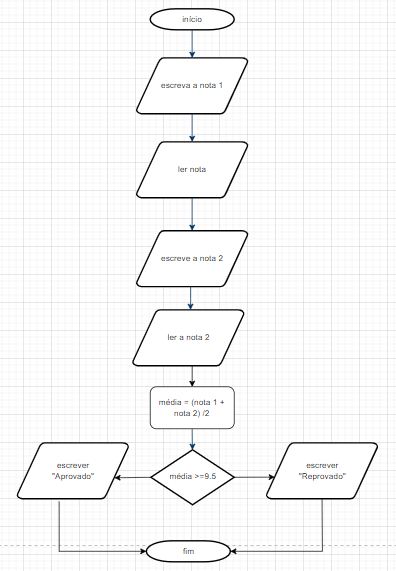
|  |
| --- |
| E01. Média aritmética |

1. Linguagem natural

Nota final

1. Obter as 2 notas dos testes do aluno
2. Calcular a média aritmética das 2 notas
3. Se a média for maior ou igual a 9.5, escreve “aprovado”
4. Caso contrário, escrever “reprovado”

2. Fluxograma (DrawIO)



3. Pseudocódigo

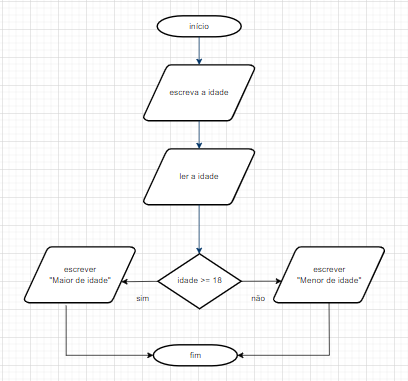
|  |
| --- |
| Algoritmo:  Calcular média final e dar avaliação  Variáveis:  nota1, real  nota2, real  media, real  INICIO  ESCREVER (“Nota1:”)  LER (nota1)  ESCREVER (Nota2:”)  LER (nota2)  Media <-- ((nota1 + nota2)/2)  SE (media >= 9.5) ENTÃO  ESCREVER (“Aprovado”)  SENAO  ESCREVER (“Reprovado”)  FIM SE  FIM |

|  |
| --- |
| E02. Maior de idade  Identificar uma pessoa maior de idade |

1. Linguagem natural

1. Obter a idade
2. Se a idade for maior ou igual a 18, escreve “Maior de idade”
3. Caso contrário, escrever “Menor de idade”

2. Fluxograma (DrawIO)



3. Pseudocódigo

|  |
| --- |
| Algoritmo:  Identificar uma pessoa maior de idade  Variáveis:  idade, inteiro  INICIO  ESCREVER (“Qual a sua idade?”)  LER (idade)  SE (idade >= 18) ENTÃO  ESCREVER (“Maior de idade.”)  SENAO  ESCREVER (“Menor de idade.”)  FIM SE  FIM |

|  |
| --- |
| E03. Números maior ou igual  Identificar se de dois números inseridos, um é maior que outro ou se é igual |

|  |
| --- |
| Algoritmo:  Identificar se de dois números inseridos, um é maior que outro ou se é igual  Variáveis:  numero1, inteiro  numero2, inteiro  INICIO  ESCREVER (“Insira o primeiro nº:”)  LER (numero1)  ESCREVER (“Insira o segundo nº:”)  LER (numero2)  SE (numero1 = numero2) ENTÃO  ESCREVER (“Os números são iguais.”)  SENAO  SE (numero1 > numero2) ENTÃO  ESCREVER (“O primeiro nº é maior.”)  SENAO  ESCREVER (“O segundo nº é maior.”)  FIM SE  FIM SE  FIM |

|  |
| --- |
| E04. Maior de idade  Identificar uma pessoa maior de idade repetir enquanto |

|  |
| --- |
| Algoritmo:  Identificar uma pessoa maior de idade repetir até  Variáveis:  idade, inteiro  INICIO  REPETIR  ESCREVER (“Qual a sua idade?”)  LER (idade)  ATÉ ( idade => 18)  ESCREVER (“ Maior de idade.”)  FIM |

|  |
| --- |
| E04. Maior de idade  Identificar uma pessoa maior de idade repetir enquanto |

|  |
| --- |
| Algoritmo:  Identificar uma pessoa maior de idade repetir enquanto  Variáveis:  idade, inteiro  INICIO  ESCREVER (“Qual a sua idade?”)  LER (idade)  REPETIR ENQUANTO ( idade < 18) FAZER  ESCREVER (“ Menor de idade.”)  ESCREVER (“Inserir novamente a idade:”)  LER (idade)  FIM-ENQUANTO  ESCREVER (“ Maior de idade.”)  FIM |

|  |
| --- |
| E05. Inscrever o formando no curso Programming Fundamentals se não souber programar, caso contrário, no de C# Foundations. |

|  |
| --- |
| Algoritmo:  Inscrever o formando no curso Programming Fundamentals se não souber programar, caso contrário, no de C# Foundations.  Variáveis:  programar, booleano  curso, texto  INICIO  ESCREVER (“Sabes programar?”)  LER (programar)  SE (programar) ENTÃO  curso <-- "C# Foundation"  SENÃO  curso <-- "Programming Fundamentals"  FIM SE  ESCREVER (“ Inscrito no curso” + curso)  FIM |

|  |
| --- |
| E06. Calcular a some de 2 números enquanto amnos diferentes de zero |

|  |
| --- |
| Algoritmo:  Calcular a some de 2 números enquanto ambos diferentes de zero  Variáveis:  a, inteiro  b, inteiro  c, inteiro  INICIO  ESCREVER (“Introduz o primeiro nº:”)  LER (a)  ESCREVER (“Introduz o segundo nº:”)  LER (b)  REPETIR ENQUANTO ( a<>0 E b<>0 ) FAZER  c <-- a+b  ESCREVE (“Soma:” + c )  ESCREVER (“Introduz o primeiro nº:”)  LER (a)  ESCREVER (“Introduz o segundo nº:”)  LER (b)  FIM-ENQUANTO  FIM |

|  |
| --- |
| E07. E06. Refatorado para subprocedimentos |

|  |
| --- |
| Algoritmo:  Calcular a some de 2 números enquanto ambos diferentes de zero  PROCEDIMENTO LerNumeros()  Variáveis:  a, inteiro  b, inteiro  ESCREVER (“Introduz o primeiro nº:”)  LER (a)  ESCREVER (“Introduz o segundo nº:”)  LER (b)  FIM-PROCEDIMENTO  FUNÇÃO CalcularSoma(a: inteiro, b: inteiro): inteiro  Variáveis:  c, inteiro  c <-- a+b  DEVOLVER c  FIM- FUNÇÃO  PROCEDIMENTO EscreverSoma(c: inteiro)  ESCREVE (“Soma:” + c )  FIM PROCEDIMENTO    INICIO  LerNumeros()  REPETIR ENQUANTO ( a<>0 E b<>0 ) FAZER  CalcularSoma(a, b)  EscreverSoma(c)  LerNumeros()  FIM-ENQUANTO  FIM |

|  |
| --- |
| E08. Pedir e listar os nomes dos alunos da nossa turma |

|  |
| --- |
| Algoritmo:  Pedir e listar os nomes dos alunos da nossa turma  Variáveis:  alunos[12]: texto  linha: inteiro  PROCEDIMENTO LerNome()  PARA linha de 0 até 11 FAZER  ESCREVER (“Introduz o teu nome:”)  LER (alunos [linha])  linha <-- linha + 1  FIM-PARA  FIM-PROCEDIMENTO  PROCEDIMENTO EscreverNome()  PARA linha de 0 até 11 FAZER  ESCREVE (“Nome:” + alunos[linha] )  linha <-- linha + 1  FIM-PARA  FIM PROCEDIMENTO    INICIO  LerNome()  EscreverNome()  FIM |

|  |
| --- |
| E09. Criar algoritmo que preencha uma matriz com a informação dos formandos da turma  - Deve ser consideradas 4 colunas para cada formando: nº interno, nome, função e email.  - Recorrendo a uma estrutura cíclica, listar para o ecrã a matriz |

|  |
| --- |
| Variáveis:  alunos[12,4]: texto  linha: inteiro  coluna: inteiro  PROCEDIMENTO LerDados()  PARA linha de 0 até 11 FAZER  ESCREVER (“Introduz o teu nº:”)  LER (alunos [linha,0])  ESCREVER (“Introduz o teu nome:”)  LER (alunos [linha,1])  ESCREVER (“Introduz a tua função:”)  LER (alunos [linha,2])  ESCREVER (“Introduz o teu email:”)  LER (alunos [linha,3])  linha <-- linha + 1  FIM-PARA  FIM-PROCEDIMENTO  PROCEDIMENTO EscreverDados()  linha <-- 0  PARA linha de 0 até 11 FAZER  PARA coluna de 0 até 3 FAZER  ESCREVE ( alunos[linha, coluna] )  coluna <-- coluna + 1  FIM-PARA  linha <-- linha + 1  FIM-PARA  FIM PROCEDIMENTO    INICIO  LerDados()  EscreverDados()  FIM |