

1- Estrutura sequencial: “Entrada → Processamento → Saída”

1.1. Algoritmo

Sequência ordenada e finita de passos executáveis para resolver um problema (computacional).

*Em computação, o conjunto de passos é limitado e deve seguir uma notação precisa (obedecer a regras de sintaxe da linguagem).

Fluxograma

Descrição de um algoritmo em notação que combina elementos gráficos e textuais.

Simbologia

Terminal



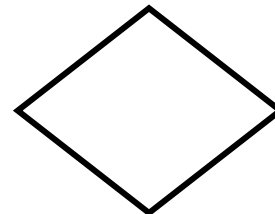
Entrada de dados via teclado



Processamento de dados (cálculos)



Tomada de decisão (condição)



Saída de dados (via vídeo)



Ponto de conexão



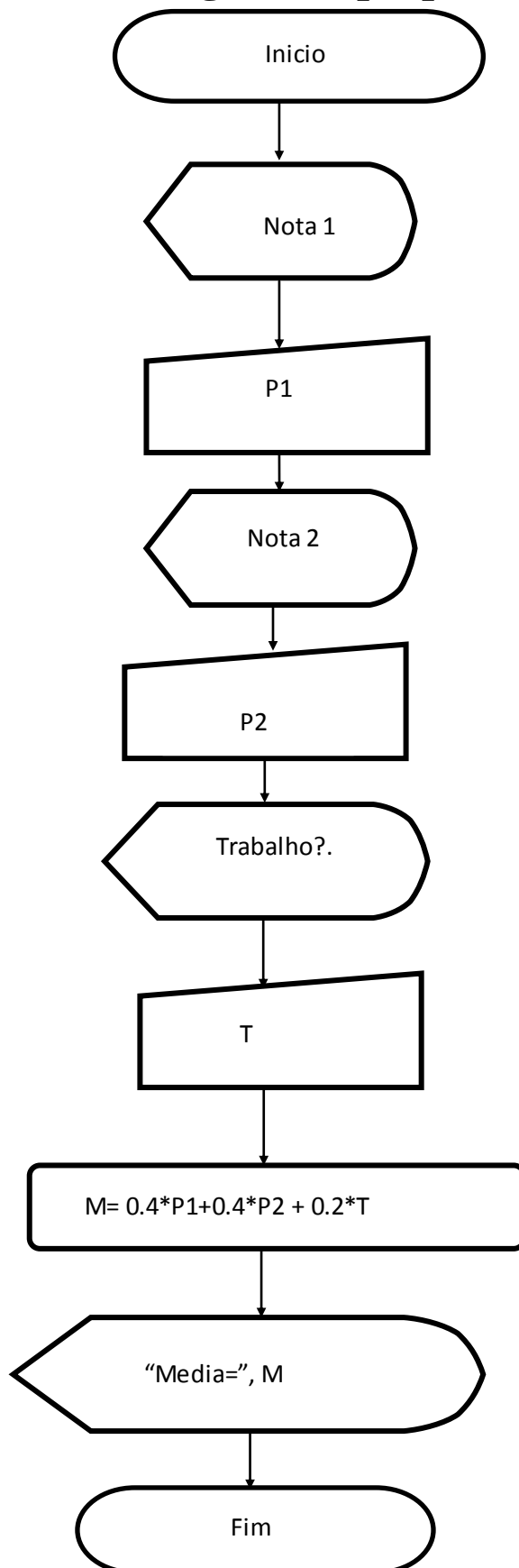
Problema proposto. Dadas duas notas de avaliações (P1, P2) e uma nota de trabalhos (T) de um aluno, calcule e informe a sua média, conforme a fórmula: $M = 0.4*P1 + 0.4*P2 + 0.2*P3$. O programa deve apresentar a seguinte tela:

Nota 1:

Nota 2:

Media =

Algoritmo proposto



1.2. **Programa.** Algoritmo descrito em linguagem de programação.

```
/*Estrutura básica para um programa em C*/
#include <stdio.h>
int main(void) {

    return 0;

}
```

Comentários

São opcionais e ignorados pelo compilador* e servem para documentação. São de dois tipos:

/* 1. Podem se estender a mais de uma linha */

// 2. Terminam no final da linha

Diretivas

Instruções ao *compilador.

Exemplo:

#include. Para incluir bibliotecas de comandos pré-definidos da linguagem. A biblioteca de entrada e saída padrão **stdio.h** (**st**andard **i**nput **o**utput **h** eader).

scanf(). Comando de entrada padrão. Obtém os dados digitados no teclado e armazena em variáveis definidas no programa;

printf(). Comando de saída padrão. Exibe dados do programa no vídeo.

***Compilador**. Programa que converte programa fonte (escrito em uma linguagem computacional) em programa executável (em linguagem de máquina)

Função principal: main().

Onde é executada a lógica do programa.

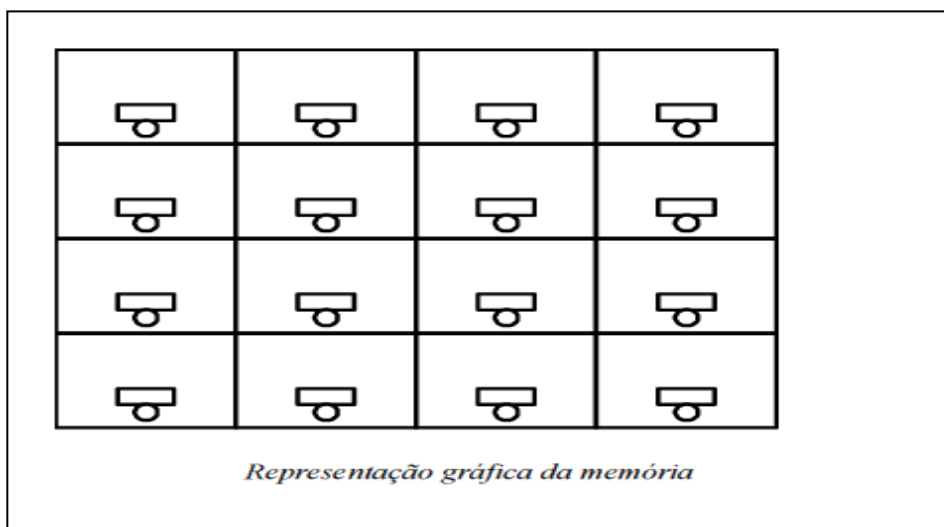
Todo o programa em C tem uma função principal onde se inicia a execução do programa. Os comando da função main() são delimitados por um par de chaves { }

Cada instrução encerra com; (ponto e vírgula) que faz parte do comando.

Como são armazenados os dados no computador?

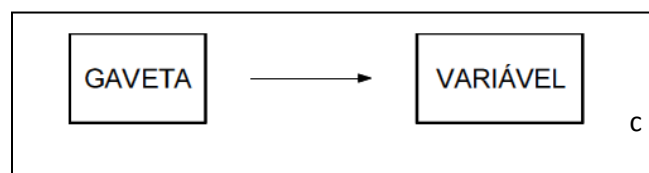
Na memória do computador.

- Imagine a memória como um grande arquivo, com várias gavetas.
- Cada gaveta possui um **nome** e consegue armazenar um único dado de cada vez.



Variáveis

- Local da memória do computador que armazena um dado.



- Devem ser declaradas antes de seu uso.
- A declaração consiste em fornecer **nome** e **tipo**.

Exemplo: **float**: P1, P2, T; **float**. Indica três variáveis para armazenamento de números reais. Não é indicada no fluxograma.

- Os nomes usados fazem referência a variáveis, funções, rótulos e vários outros objetos definidos pelo usuário.
- primeiro caractere deve ser uma letra ou um sublinhado.
- Os 32 primeiros caracteres de um identificador são significativos.
- É case sensitive, ou seja, as letras maiúsculas diferem das minúsculas. **int Fatec; /*é diferente de int fatec;*/**

Nomes ou identificadores consideram as seguintes regras:

- No máximo 32 caracteres;
- Devem começar por letra, ou o caractere de sublinha;
- Podem conter letras, números e o caractere sublinha;
- Não podem ser o nome de uma palavra reservada.

Palavras reservadas:

Auto	double	if	static	break
case	char	continue	default	do
extern	entry	else	float	for
goto	int	long	register	short
sizeof	struct	switch	typedef	return
union	unsigned	while		

Solução do problema proposto

```
/* media.c. Calcula a média de notas de um aluno */

#include <stdio.h>
int main(void) {
    float P1, P2, T, M;
    printf("\nNota1: ");          scanf("%f", &P1);
    printf("\nNota2: ");          scanf("%f", &P2);
    printf("\nTrabalho: ");       scanf("%f", &T);
    M=0.4*P1+0.4*P2+0.2*T;
    printf("\nMedia = %.1f\n", M);

return 0;
}
```

Complementos

- **printf("\nNota1: ");** Impressão de dados formatados. Imprime no vídeo a palavra **Nota1**: após pular uma linha (\n é um comando para pular uma linha. New line).
- **scanf("%f", &T);** Leitura de dado formatados do teclado. Leitura da variável T do tipo float (real). &T indica o endereço da variável T na memória do computador.

Expressões aritméticas.

Compostas de constantes (Exemplo: 0.40, 0.20,..) , variáveis (Exemplos: P1, P2, T, M) e operadores aritméticos (+ - * /). O operador de atribuição (=) avalia a expressão a sua direita e armazena o resultado na variável a sua esquerda.

return 0. Encerra a execução da função main().

1.3. printf() – COMANDO EXIBIR/IMPRIMIR no vídeo

Sintaxe: `printf("expressão", lista de argumentos);`

a) expressão: É a formatação do valor a ser lido.

b) lista de argumentos: Consta da variável e o seu endereço. O símbolo `&` na frente da variável, retorna o endereço da variável.

Exemplo 1

```
#include <stdio.h>
Int main(void) {
int idade; idade = 50;
printf("Minha idade é %d", idade);
return 0;    }
}
```

Saída: Minha idade é 50

// Exemplo 2

```
#include <stdio.h>
Int main(void) {
int evento; char corrida; float tempo;
evento = 5; corrida = 'C'; tempo = 27.25;
printf("O tempo vitorioso na eliminatória %c", corrida);
printf("\nda competicao %d foi %.2f.", evento, tempo);
return 0;    }
```

Saída:

O tempo vitorioso na eliminatória C
da competição 5 foi 27.25.

Obs: A variável tempo do tipo *float* foi formatada com 2 casas decimais.

// exemplo 3

```
#include <stdio.h>
Int main (void)
{
printf("Minha idade é %d.", 20);
return 0;    }
```

Saída: Minha idade é 20.

// exemplo 4

```
#include <stdio.>
```

```
int main(void)
```

```
{ printf("%s está a %d milhões de milhas\ndo sol", "Vênus", 67);
```

```
return 0; }
```

Saída: Vênus está a 67 milhões de milhas do sol

CÓDIGO PARA IMPRESSÃO FORMATADA DO COMANDO printf():

CÓDIGO printf() FORMATO

 %c

 Caractere simples

 %d

 Decimal

 %f

 Ponto flutuante

 %x

 Hexadecimal

Exemplo 5

```
#include <stdio.h>
```

```
int main(void) {
```

```
printf("Ana\tBete\tCarlos\n");
```

```
printf("Lista de convidados: ");
```

```
printf("\n");
```

```
return 0; }
```

Saída:

Ana Bete Carlos

“Lista de convidados”

// Exemplo 6. Tamanho mínimo para a impressão de um campo

```
#include <stdio.h>
```

```
int main(void) {
```

```
printf("Os alunos são %2d.\n", 80);
```

```
printf("Os alunos são %4d.\n", 80);
```

```
printf("Os alunos são %6d.\n", 80);
```

```
return 0;
```

```
/* Espera pressionar uma tecla */
```

```
}
```

Saída:

Os alunos são 80.

Os alunos são 80.

Os alunos são 80.

Exemplo 7. FORMATANDO NÚMEROS EM PONTO FLUTUANTES

```
#include <stdio.h>
int main(void)    {
printf("Valor = %7.2f\n", 1234.76);
printf("Valor = %3.2f\n", 1234.76);
printf("Valor = %7.1f\n", 1234.74);
return 0; /* Espera pressionar uma tecla */
}
```

Saída:

Valor = 1234.76

Valor = 1234.76

Valor = 1234.7

%a.b

a total de casas (conta tudo: inteiro, ponto e casa decimal)

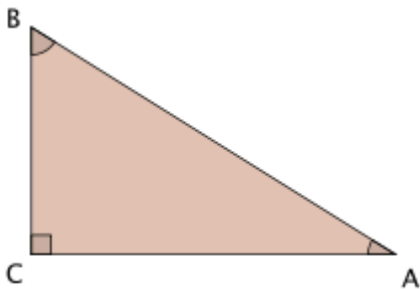
b casas decimais.

1.5. Exercícios

1.1). Faça um programa que converte a temperatura de Fahrenheit para Celsius. Fórmula para conversão de grau Fahrenheit para grau Celsius. $^{\circ}\text{C} = (^{\circ}\text{F} - 32)/1,8$.

1.2). Faça um programa que leia a altura e a base de um triângulo. Calcule e exiba a sua área (S) onde h é a altura do triângulo, b a medida da base. ($S = (b * h)/2$).

1.3). Faça um programa que receba a medida de um triângulo retângulo. (cateto oposto, cateto adjacente e hipotenusa: a , b , c). Calcule e exiba o seno e o cosseno do ângulo A .



$\begin{aligned} \text{sen } A &= \text{cateto oposto/hipotenusa;} \\ \text{cos } A &= \text{cateto adjacente/hipotenusa} \end{aligned}$
--

1.4) Faça um programa que leia pelo teclado um valor, em dólares, converta e exiba o mesmo valor em reais. Considere que U\$ 1,00 seja equivalente a R\$ 3,50.

1.5) Sabe-se, pela física, que um corpo em movimento retilíneo uniformemente variado segue a seguinte equação:

$$S = s_0 + v_0 * t + \frac{1}{2} * a * t^2;$$

Faça um programa que leia as medidas de s_0 (em m), v_0 (em m/s), t (em s) e a (em m/s^2), calcule e exiba a posição S em que se encontra tal corpo. Exemplificando, se as entradas fossem $s_0 = 2.0\text{m}$, $v_0 = 3.0\text{m/s}$, $t = 4\text{s}$ e $a = 10 \text{ m/s}^2$, então a saída seria 94.0.

1.6) Receber as medidas de uma sala em metros (c , l), bem como o preço do carpete por metro quadrado (ppm), informe o custo total para forrar o piso da sala.

1.7. Dado o raio de uma esfera em metros (r): Faça um programa que receba r e informe:

a- o volume ($\frac{4}{3} * \text{PI} * R^3$) e a superfície ($4 * \text{PI} * R^2$) da mesma.

b - Supondo que a esfera seja de ferro maciço (densidade = $7,8 \text{ g/cm}^3$), forneça a massa da esfera.

O programa deve seguir a seguinte tela:

Informe o raio da esfera: _____<enter>

O volume da esfera e: _____ A área da esfera e: ----- A massa da esfera e: _____

2. Estruturas sequenciais. (Continuação)

2.1. Tipos de dados, constantes e variáveis.

Constantes. Valores inalterados na execução do programa. Exemplos:

Constantes do tipo inteiro: 32, 45,..

Constantes do tipo real: 32.01, 45,2345...

Constantes do tipo caractere: 'a', 'b', ..

Constantes do tipo cadeia de caractere (string): "abc", "fatec",...

Variáveis:

Valores que podem ser alterados na execução do programa;
Representa uma posição de memória que armazena um dado específico identificado por um nome;
Os nomes de variáveis são iniciados por letra e contendo letras, números e o símbolo _.

Exemplos válidos: A1,..C1, fatec_sp,..

O tipo da variável (int, float, char,..) deve ser declarado antes de ser usada em um programa.

Palavras reservadas:

Auto	double	if	static	break
case	char	continue	default	do
extern	entry	else	float	for
goto	int	long	register	short
sizeof	struct	switch	typedef	return
union	unsigned	while		

2.2. Operadores

Símbolos usados na criação de expressões (aritméticas, lógicas, comparação).

Operadores aritméticos: + - * / (inteira e real) % (resto da divisão).

Exemplos:

Operação de atribuição: Obtém o valor do seu lado direito e atribui à variável do lado esquerdo. Ex. $a=b+3$; Soma o valor da variável b com o valor constante 3 e atribui a variável a ;

$A= 5$; $A=A+32$; O valor final de A será 37.