INSTITUTO FEDERAL
Paraná
Campus Paranaguá

Ministério da Educação

ENSINO MÉDIO INTEGRADO - INFORMÁTICA
Disciplina de Linguagem de Programação
Aula 02: Comandos de Entrada e Saída

Gil Eduardo de Andrade

Conceitos Preliminares

(https://docs.microsoft.com/pt-br/cpp/c-language/?view=msvc-170)

Introdução

Os comandos de entrada e saída permitem ao programador interagir com o usuário, possibilitando que informações sejam enviadas e recebidas do usuário.

Comando de Saída

O principal e mais utilizado comando de saída, na linguagem C, é o **printf()** - visto na aula anterior. O **printf()** permite o envio de textos estáticos e também o valor de variáveis ao monitor do computador (saída padrão). Exemplo: **printf("Gil Eduardo de Andrade");**

- Caracteres Especiais:
  - **"\n":** permite que quebras de linha sejam efetuadas na utilização do comando **printf()**. Exemplo: **printf("Gil\nEduardo\nAndrade\n");**

  - **"\t":** permite que tabulações sejam criadas na utilização do comando **printf()**. Exemplo: **printf("\tGil\n\t\tEduardo\n\t\t\tAndrade");**

- Exibição de Conteúdo das Variáveis:
  - **"%i" ou "%d":** permite exibir o conteúdo das variáveis do tipo **int** na utilização do **printf()**. Exemplo: **printf("%i", soma);**

  - **"%c":** permite exibir o conteúdo das variáveis do tipo **char** na utilização do **printf()**. Exemplo: **printf("%c", letra);**

- **"%f":** permite exibir o conteúdo das variáveis do tipo **float** na utilização do **printf()**. Exemplo: **printf("%f", pi);**

Comando de Entrada

os comandos de entrada permitem que informações especificadas pelo usuário sejam recebidas pelo programa codificado. O principal e mais utilizado comando de entrada, na linguagem C, é o **scanf()**. O **scanf()** permite ao programador ler informações fornecidas pelo usuário em tempo de execução, através do teclado (entrada padrão). Exemplo: **scanf("%i", &valor);**

- **"%i":** para obter um valor inteiro;
- **"%c":** para obter um caractere;
- **"%f":** para obter um valor float;
- **"%d":** para obter um valor decimal.

Lendo vários valores com apenas um scanf()

Também é possível efetuar a leitura de vários valores através da utilização de apenas um comando **scanf()**. Os múltiplos valores de entrada podem ou não ser do mesmo tipo. Exemplo: **scanf("%i %c %f", &inteiro, &letra, &flutuante);**

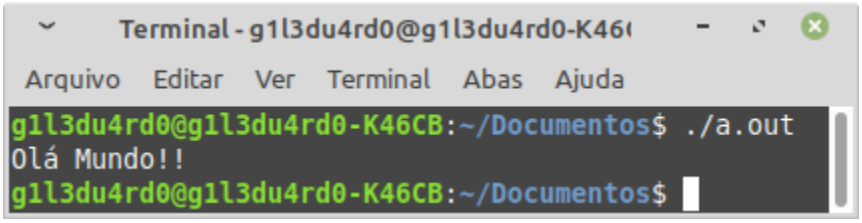
Comando getchar()

O comando **getchar()** permite efetuar a leitura de caracteres de forma análoga ao comando scanf("%c"). Entretanto, sua sintaxe é mais simples, pois é preciso apenas atribuí-lo a variável que receberá o caractere digitado. Exemplo: **letra = getchar();**

Codificação – Linguagem de Programação C

Comando de Saída - printf(“”);

```
#include <stdio.h>

int main() {

   printf("Olá Mundo!!");

   return 0;

}



                        Comando de Saída - printf(“”);

          ____________________________________________________________

       [printf(“Olá Mundo”);] - exibe o texto “Olá Mundo” no monitor/terminal

          ____________________________________________________________

                   (Arquivo-fonte: 01 - Introdução C/printf.c)
```



Resultado da execução do código anterior

Comando de Saída - Caracteres Especiais “\n”

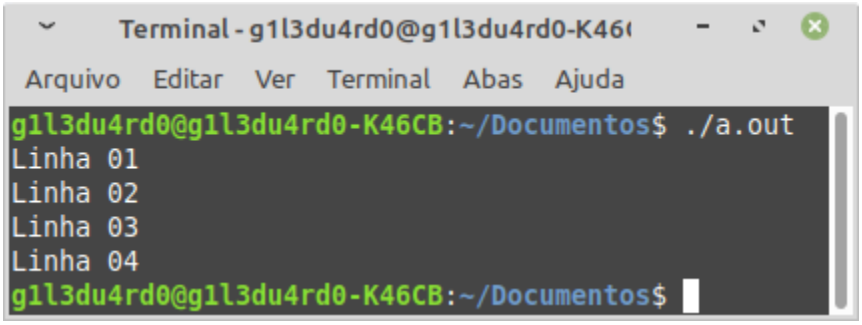
```
#include <stdio.h>

int main() {
```

```
    printf("Linha 01\nLinha 02\nLinha 03\nLinha 04");

    return 0;

}
```

Comando de Saída - “\n”

[printf(“\n”);] - permite quebra linha dentro do texto inserido no printf();

(Arquivo-fonte: 01 - Introdução C/printf_barran.c)



Resultado da execução do código anterior

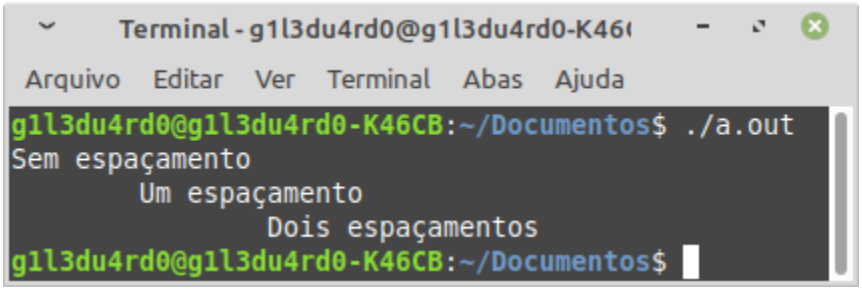
Comando de Saída - Caracteres Especiais “\t”

```
#include <stdio.h>

int main() {

    printf("Sem espaçamento\n\tUm espaçamento\n\t\tDois espaçamentos");

    return 0;

}
```

Comando de Saída - “\t”

[printf(“\n”);] - permite adicionar tabulações dentro do texto inserido no printf();

(Arquivo-fonte: 01 - Introdução C/printf_barrat.c)



Resultado da execução do código anterior

Comando de Saída - Exibindo o Conteúdo das Variáveis

```c
#include <stdio.h>

int main() {

   int a = 10;

   float b = 3.1415;

   char c = 'G';

   printf("Valor da variável a: %i (int)", a);

   printf("\nValor da variável b: %f (float)", b);

   printf("\nValor da variável c: %c (char)", c);

   return 0;

}
```

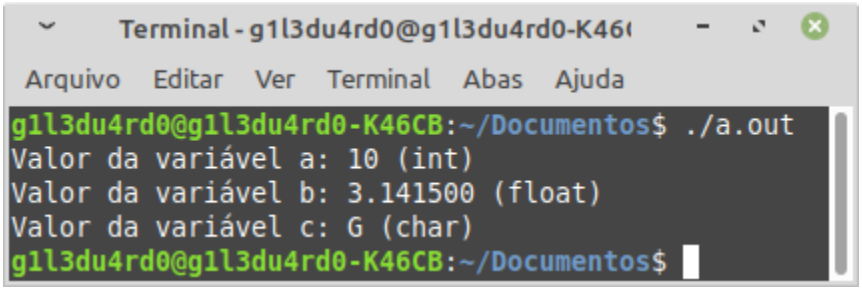
Comando de Saída - “\t”

[printf(“%i”, a);] - exibe o conteúdo da variável “a” do tipo “int”

[printf(“%f”, b);] - exibe o conteúdo da variável “b” do tipo “float”

[printf(“%c”, c);] - exibe o conteúdo da variável “c” do tipo “char”

(Arquivo-fonte: 01 - Introdução C/printf_barrat.c)

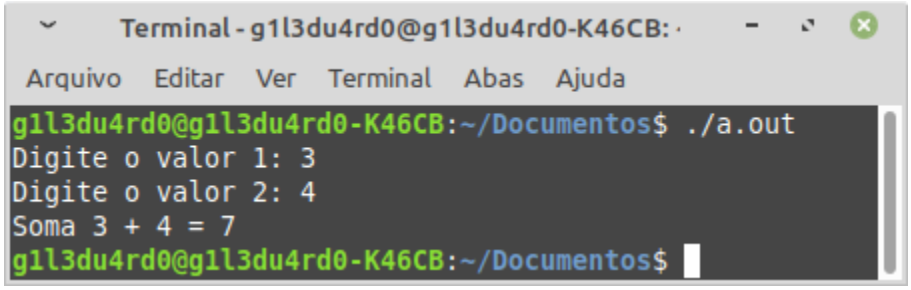


Resultado da execução do código anterior

Comando de Entrada “scanf()” - Inteiros

```
#include <stdio.h>

int main() {

   int v1, v2, soma;

   printf("Digite o valor 1: ");

   scanf("%i", &v1);

   printf("Digite o valor 2: ");

   scanf("%i", &v2);

   soma = v1 + v2;

   printf("Soma %i + %i = %i", v1, v2, soma);

   printf("\n");

   return 0;

}
```

```
                    Comando de Entrada - “scanf(“%i”)”
          ____________________________________________________________
    [scanf(“%i”, &v1);] - obtém o valor inteiro digitado e armazena na variável v1
    [scanf(“%i”, &v2);] - obtém o valor inteiro digitado e armazena na variável v2
          ____________________________________________________________
              (Arquivo-fonte: 02 - Entrada e Saída/es_scanf_int.c)
```



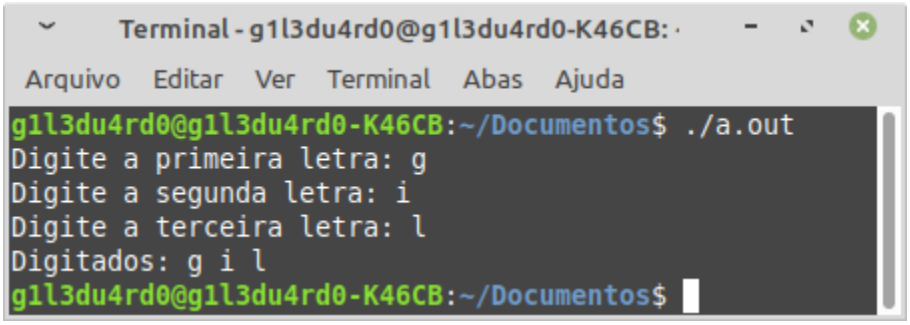
Resultado da execução do código anterior

Comando de Entrada “scanf()” - Caracteres / Espaço

```
#include <stdio.h>

int main() {

   char c1, c2, c3;

   printf("Digite a primeira letra: ");

   scanf(" %c", &c1);

   printf("Digite a segunda letra: ");

   scanf(" %c", &c2);

   printf("Digite a terceira letra: ");

   scanf(" %c", &c3);

   printf("Digitados: %c %c %c", c1, c2, c3);

   printf("\n");

   return 0;

}
```

```
                    Comando de Entrada - “scanf(“%c”)”
          ____________________________________________________________
     [scanf(“%c”, &c1);] - obtém o caractere digitado e armazena na variável c1
     [scanf(“%c”, &c2);] - obtém o caractere digitado e armazena na variável c2
     [scanf(“%c”, &c3);] - obtém o caractere digitado e armazena na variável c2
```

(Arquivo-fonte: 02 - Entrada e Saída/es_scanf_char_espaco.c)

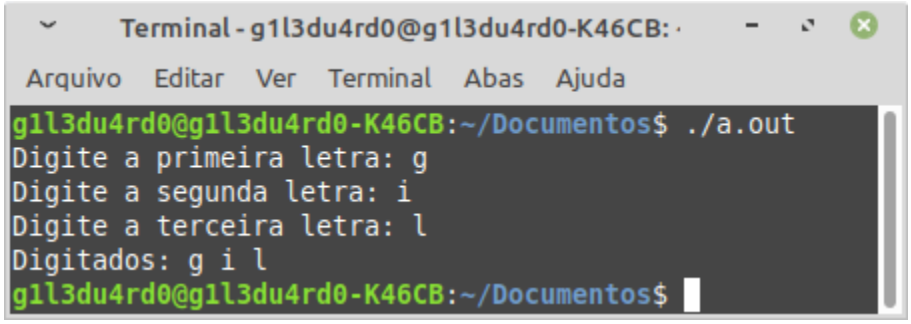


Resultado da execução do código anterior

Comando de Entrada "scanf()" - Caracteres / __fpurge(stdin)

```
#include <stdio.h>

#include <stdio_ext.h>

int main() {

   char c1, c2, c3;

   printf("Digite a primeira letra: ");

   __fpurge(stdin); // fflush(stdin);

   scanf("%c", &c1);

   printf("Digite a segunda letra: ");

   __fpurge(stdin);

   scanf("%c", &c2);

   printf("Digite a terceira letra: ");

   __fpurge(stdin);

   scanf("%c", &c3);

   printf("Digitados: %c %c %c", c1, c2, c3);
```

```
    printf("\n");

    return 0;

}


                Comando de Entrada - “scanf(“%c”)” + “__fpurge(stdin)”

            ____________________________________________________________

[__fpurge(stdin);] - limpa o buffer e garante a execução adequada do próximo “scanf()”

            ____________________________________________________________

             (Arquivo-fonte: 02 - Entrada e Saída/es_scanf_char_fpurge.c)
```

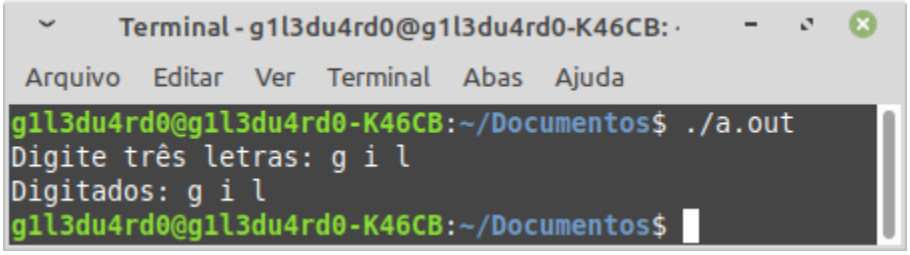


Resultado da execução do código anterior

Comando de Entrada “scanf()” - Caracteres / Juntos

```
#include <stdio.h>

int main() {

    char c1, c2, c3;

    printf("Digite três letras: ");

    scanf("%c %c %c", &c1, &c2, &c3);

    printf("Digitados: %c %c %c", c1, c2, c3);

    printf("\n");

    return 0;
```

```
}

                    Comando de Entrada - “scanf(“%c %c %c”)”
          ____________________________________________________________
  [scanf(“%c %c %c”, &c1, &c2, &c3);] - obtém os caracteres digitados e armazena nas
                              variáveis c1, c2 e c3
          ____________________________________________________________
            (Arquivo-fonte: 02 - Entrada e Saída/es_scanf_char_junto.c)
```

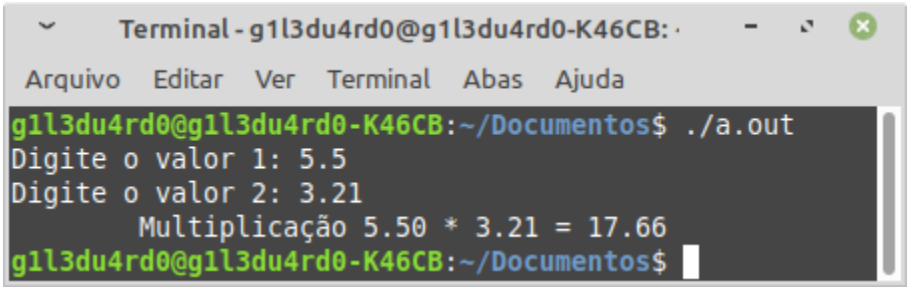


Resultado da execução do código anterior

Comando de Entrada “scanf()” - Ponto Flutuante

```
#include <stdio.h>

int main() {

   float v1, v2, mult;

   printf("Digite o valor 1: ");

   scanf("%f", &v1);

   printf("Digite o valor 2: ");

   scanf("%f", &v2);

   mult = v1 * v2;

   printf("\tMultiplicação %.2f * %.2f = %.2f", v1, v2, mult);

   printf("\n");

   return 0;
```

```
}

                    Comando de Entrada - “scanf(“%f”)”
        ____________________________________________________________
      [scanf(“%f”, &v1);] - obtém o valor digitado e armazena na variável v1
      [scanf(“%f”, &v2);] - obtém o valor digitado e armazena na variável v2
        ____________________________________________________________
            (Arquivo-fonte: 02 - Entrada e Saída/es_scanf_float.c)
```

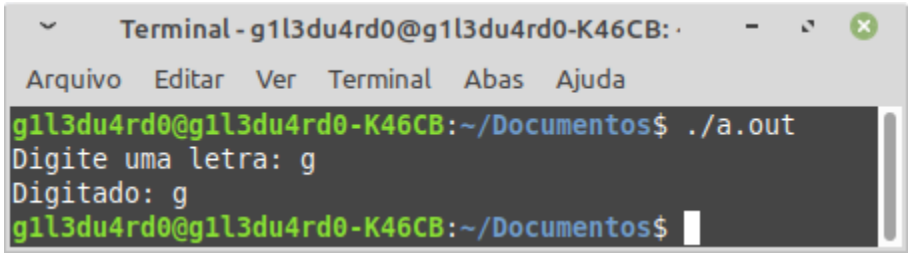


Resultado da execução do código anterior

Comando de Entrada “getchar()” - Variável

```
#include <stdio.h>

int main() {

   char c1;

   printf("Digite uma letra: ");

   c1 = getchar();

   printf("Digitado: %c", c1);

   printf("\n");

   return 0;

}

                    Comando de Entrada - “getchar()” / Variável
```

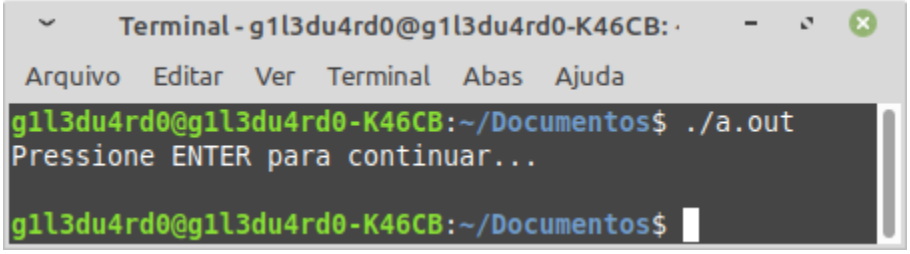
```
_____________________________________________________________

[c1 = getchar();] - obtém o caractere digitado e armazena na variável c1

_____________________________________________________________

(Arquivo-fonte: 02 - Entrada e Saída/es_getchar_variavel.c)
```



Resultado da execução do código anterior

Comando de Entrada “getchar()” - Enter

```
#include <stdio.h>

int main() {

   printf("Pressione ENTER para continuar...");

   getchar();

   printf("\n");

   return 0;

}


                    Comando de Entrada - “getchar()” / Enter

_____________________________________________________________

[getchar();] - Aguarda a tecla ENTER ser pressionada

_____________________________________________________________

(Arquivo-fonte: 02 - Entrada e Saída/es_getchar_enter.c)
```



Resultado da execução do código anterior