

Trabalho 01

O trabalho possui diversas funcionalidades, tais como ignorar certas entradas como espaços em branco, tabulações e quebras de linha. Ele também encerra suas linhas quando encontra um ponto e vírgula. Além disso, ele ignora comentários que estão após a marcação "//".

O programa permite realizar incrementos e decrementos utilizando os operadores ++ e --. Também contém uma função chamada "Function" e comandos de decisão, como "if", "else", e "elseif". Para repetições, foram utilizados os comandos "while" e "do while".

O programa é capaz de trabalhar com números inteiros [0, 9], letras minúsculas [a, z], letras maiúsculas [A, Z] e os operadores lógicos >, <, == e !=, além dos operadores aritméticos +, -, *, e /. Para a atribuição de valores, é utilizado apenas um sinal de igual (=).

Por fim, o programa também é capaz de trabalhar com variáveis booleanas (V e F), números inteiros (int), números com casas decimais (float e double), caracteres (char), bytes (byte) e strings (texto).

Dificuldades:

Devido ao fato de estar cursando três disciplinas no mesmo horário, perdi parte do conteúdo e precisei estudar o dobro em casa. A maior dificuldade encontrada foi durante a instalação do Eclipse, pois houve um erro e a situação se complicou quando tentei arrumar, já que conflitou com um dos programas utilizados no trabalho. Como solução, precisei utilizar um computador mais antigo da família, o que atrasou ainda mais o trabalho. No entanto, apesar dos contratempos, consegui aprender bastante e não tive grandes dificuldades.

Código txt

```
boolean a = true;  
double b = 2, c = 0;
```

```
int a1 = 2, b1 = 3;  
double c1, d, e;
```

```
while (k < 10) {  
  while (j < 5) {  
    j++;  
    if (j == 4) {  
      j = j + 28;  
    }  
    int o = 11;  
    k = j + 5.5;  
    j = j + 1;
```

```
k++;  
}  
}
```

```
if (a == 1) {  
    a = 2;  
}  
if(a != 1){  
    a = 2;  
}else{  
    a = 3;  
}
```

```
if(a > 15){  
    b = 32;  
}elseif(b < 2){  
    b = 27;  
}
```

```
if(a != 1){  
    c = 4;  
}elseif(b == 3){  
    b = 500;  
}else{  
    a = 315;  
}
```

```
do{  
    a=1+1;
```

```
}while(a<14)
```

```
function () {  
    a = x + 1;  
}  
function (int a, int b) {  
    a = x + 1;  
}
```

```
function (int a) {  
    a = x + 1;  
}
```

```
switch(b){  
    case 1:  
        b = b+12;  
        break;  
    case 2:
```

```

        b = b+22;
        break;
}

```

Imagens do Console

```

Lendo arquivo fonte...
Call: Start
  Call: COMANDOS
    Call: IDENTIFICADOR
      Call: DECLARAR_VARIAVEL
        Consumed token: <<TIPO>>: "boolean" at line 1 column 1>
        Consumed token: <<LITERALS>>: "a" at line 1 column 9>
      Return: DECLARAR_VARIAVEL
        Consumed token: <"=" at line 1 column 11>
      Call: ATRIBUICAO_VARIAVEL
        Consumed token: <<BOOL>>: "true" at line 1 column 13>
      Return: ATRIBUICAO_VARIAVEL
        Consumed token: <";" at line 1 column 17>
    Return: IDENTIFICADOR
  Call: IDENTIFICADOR
    Call: DECLARAR_VARIAVEL
      Consumed token: <<TIPO>>: "double" at line 2 column 1>
      Consumed token: <<LITERALS>>: "b" at line 2 column 8>
    Return: DECLARAR_VARIAVEL
      Consumed token: <"=" at line 2 column 10>
    Call: ATRIBUICAO_VARIAVEL
      Consumed token: <<VALUES>>: "2" at line 2 column 12>
    Return: ATRIBUICAO_VARIAVEL
      Consumed token: <"," at line 2 column 13>
      Consumed token: <<LITERALS>>: "c" at line 2 column 15>
      Consumed token: <"=" at line 2 column 17>
    Call: ATRIBUICAO_VARIAVEL
      Consumed token: <<VALUES>>: "0" at line 2 column 19>
    Return: ATRIBUICAO_VARIAVEL
      Consumed token: <";" at line 2 column 20>
  Return: IDENTIFICADOR
Call: IDENTIFICADOR
  Call: DECLARAR_VARIAVEL
    Consumed token: <<TIPO>>: "int" at line 4 column 1>
    Consumed token: <<LITERALS>>: "a1" at line 4 column 5>
  Return: DECLARAR_VARIAVEL
    Consumed token: <"=" at line 4 column 8>
  Call: ATRIBUICAO_VARIAVEL
    Consumed token: <<VALUES>>: "2" at line 4 column 10>
  Return: ATRIBUICAO_VARIAVEL
    Consumed token: <"," at line 4 column 11>
    Consumed token: <<LITERALS>>: "b1" at line 4 column 13>
    Consumed token: <"=" at line 4 column 16>
  Call: ATRIBUICAO_VARIAVEL
    Consumed token: <<VALUES>>: "3" at line 4 column 18>
  Return: ATRIBUICAO_VARIAVEL
    Consumed token: <";" at line 4 column 19>
Return: IDENTIFICADOR
Call: IDENTIFICADOR
  Call: DECLARAR_VARIAVEL
    Consumed token: <<TIPO>>: "double" at line 5 column 1>
    Consumed token: <<LITERALS>>: "c1" at line 5 column 8>
  Return: DECLARAR_VARIAVEL

```

```

Return: IDENTIFICADOR
Call: WHILE
  Consumed token: <"while" at line 7 column 1>
  Consumed token: <"(" at line 7 column 7>
  Call: EXP_BOOL
    Consumed token: <<LITERALS>: "k" at line 7 column 8>
    Consumed token: <<OPER_RELACIONAL>: "<" at line 7 column 10>
    Consumed token: <<VALUES>: "10" at line 7 column 12>
  Return: EXP_BOOL
  Consumed token: <")" at line 7 column 14>
  Call: ABRE_ESCOPO
    Consumed token: <"{" at line 7 column 16>
    Call: COMANDOS
      Call: WHILE
        Consumed token: <"while" at line 8 column 1>
        Consumed token: <"(" at line 8 column 7>
        Call: EXP_BOOL
          Consumed token: <<LITERALS>: "j" at line 8 column 8>
          Consumed token: <<OPER_RELACIONAL>: "<" at line 8 column 10>
          Consumed token: <<VALUES>: "5" at line 8 column 12>
        Return: EXP_BOOL
        Consumed token: <")" at line 8 column 13>
        Call: ABRE_ESCOPO
          Consumed token: <"{" at line 8 column 15>
          Call: COMANDOS
            Call: EXP_ARITMETICA
              Consumed token: <<LITERALS>: "j" at line 9 column 1>
              Consumed token: <<INC_DEC>: "++" at line 9 column 2>
              Consumed token: <";" at line 9 column 4>
            Return: EXP_ARITMETICA
            Call: IF
              Consumed token: <"if" at line 10 column 1>
              Call: CONDICAO
                Consumed token: <"(" at line 10 column 4>
                Call: EXP_BOOL
                  Consumed token: <<LITERALS>: "j" at line 10 column 5>
                  Consumed token: <<OPER_RELACIONAL>: "==" at line 10 column 7>
                  Consumed token: <<VALUES>: "4" at line 10 column 10>
                Return: EXP_BOOL
                Consumed token: <")" at line 10 column 11>
                Consumed token: <"{" at line 10 column 13>
                Call: COMANDOS
                  Call: EXP_ARITMETICA
                    Consumed token: <<LITERALS>: "j" at line 11 column 1>
                    Consumed token: <"=" at line 11 column 3>
                    Consumed token: <<LITERALS>: "j" at line 11 column 5>
                    Consumed token: <<OPER_ARITMETICO>: "+" at line 11 column 7>
                    Consumed token: <<VALUES>: "28" at line 11 column 9>
                    Consumed token: <";" at line 11 column 11>
                  Return: EXP_ARITMETICA
                Return: COMANDOS
                Consumed token: <"}" at line 12 column 1>
              Return: CONDICAO
            Return: COMANDOS
          Return: ABRE_ESCOPO
        Return: WHILE
      Return: COMANDOS
    Return: ABRE_ESCOPO
  Return: WHILE
Return: IDENTIFICADOR

```

```

        Consumed token: < ; at line 56 column 18>
    Return: EXP_ARITMETICA
    Return: COMANDOS
    Consumed token: <">" at line 57 column 1>
    Return: ABRE_ESCOPO
Return: FUNCAO
Call: SWITCH_CASE
    Consumed token: <"switch" at line 59 column 1>
    Consumed token: <"(" at line 59 column 7>
    Consumed token: <<LITERALS>: "b" at line 59 column 8>
    Consumed token: <">" at line 59 column 9>
    Consumed token: <"{" at line 59 column 10>
    Call: CASE
        Consumed token: <"case" at line 60 column 9>
        Consumed token: <<VALUES>: "1" at line 60 column 14>
        Consumed token: <":" at line 60 column 15>
        Call: COMANDOS
            Call: EXP_ARITMETICA
                Consumed token: <<LITERALS>: "b" at line 61 column 17>
                Consumed token: <"=" at line 61 column 19>
                Consumed token: <<LITERALS>: "b" at line 61 column 21>
                Consumed token: <<OPER_ARITMETICO>: "+" at line 61 column 22>
                Consumed token: <<VALUES>: "12" at line 61 column 23>
                Consumed token: <";" at line 61 column 25>
            Return: EXP_ARITMETICA
        Return: COMANDOS
        Consumed token: <"break" at line 62 column 17>
        Consumed token: <";" at line 62 column 22>
    Return: CASE
    Call: CASE
        Consumed token: <"case" at line 63 column 9>
        Consumed token: <<VALUES>: "2" at line 63 column 14>
        Consumed token: <":" at line 63 column 15>
        Call: COMANDOS
            Call: EXP_ARITMETICA
                Consumed token: <<LITERALS>: "b" at line 64 column 17>
                Consumed token: <"=" at line 64 column 19>
                Consumed token: <<LITERALS>: "b" at line 64 column 21>
                Consumed token: <<OPER_ARITMETICO>: "+" at line 64 column 22>
                Consumed token: <<VALUES>: "22" at line 64 column 23>
                Consumed token: <";" at line 64 column 25>
            Return: EXP_ARITMETICA
        Return: COMANDOS
        Consumed token: <"break" at line 65 column 17>
        Consumed token: <";" at line 65 column 22>
    Return: CASE
    Consumed token: <">" at line 66 column 1>
    Return: SWITCH_CASE
Return: COMANDOS
Consumed token: <<EOF> at line 66 column 1>
Return: Start
Arquivo fonte processado corretamente!

```