

Lista 3– LP 1 – Vetores - Prof. Marcio

Entrega: 6 de abril de 2021

1. Faça um algoritmo que leia um vetor  $N[20]$ . A seguir, encontre o menor elemento do vetor  $N$  e a sua posição dentro do vetor. Exemplo de saída: “O menor elemento de  $N$  é”,  $M$ , “e sua posição dentro do vetor é:”,  $P$ .
2. Faça um algoritmo que leia um vetor  $K[30]$ . Troque a seguir, todos os elementos de **índice** ímpar do vetor com os elementos de **índice** par imediatamente posteriores.
3. Faça um algoritmo que leia um vetor  $S[20]$  e uma variável  $A$ . A seguir, mostre o produto da variável  $A$  pelo vetor.
4. Escreva um algoritmo que leia e mostre um vetor de 20 números. A seguir, conte quantos valores pares existem no vetor.
5. Escreva um algoritmo que leia um vetor de 100 posições e mostre-o ordenado, em ordem decrescente.
6. Escrever um algoritmo que gere os 30 primeiros números primos a partir de 100 e os armazena em um vetor de  $X[100]$  escrevendo, no final, o vetor  $X$ .
7. Escrever um algoritmo que lê um vetor  $K(15)$ . Crie, a seguir, um vetor  $P$ , que contenha todos os números primos de  $K$ . Escreva o vetor  $P$ .
8. Escrever um algoritmo que lê um vetor  $X(20)$ . Escreva, a seguir, cada um dos valores distintos que aparecem em  $X$  dizendo quantas vezes cada valor aparece em  $X$ .
9. Faça um algoritmo que leia um código numérico inteiro e um vetor de 50 posições de números. Se o código for zero, termine o algoritmo. Se o código for 1, mostre o vetor na ordem como ele foi lido. Se o código for 2, mostre o vetor na ordem inversa, do último até o primeiro.
10. Faça um algoritmo que leia um vetor de 10 posições. Mostre então os 3 menores valores do vetor.
11. Faça um algoritmo que leia um vetor  $A[10]$ . Preencha então um vetor  $B[10]$  com o fatorial de cada valor de  $A$  respeitando as posições, caso o referido valor for positivo ou nulo. Deixe os valores negativos intactos. Mostre o vetor  $B$ .
12. Faça um algoritmo que leia um vetor  $A[10]$ . Inverta, então, os valores de  $A$ , trocando o primeiro pelo último, o segundo pelo penúltimo e assim por diante. Mostre o vetor  $A$  após as alterações.

0

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

```
program v1;  
var  
  V: array[1..20] of integer;  
  n, i: Integer;  
begin  
  for i:=1 to 20 do  
    begin  
      WriteLn('Dê os valores de v?');  
      ReadLn(n);  
      begin  
        Write(V[i]);  
        If V[i] <= 0 then  
          WriteLn('O menor elemento de v é', n, '  
            e sua posição no vetor é P');  
        end  
      end  
    end  
  end.  
end.
```

→

:

=

;

.

[

]

:







'

(


)

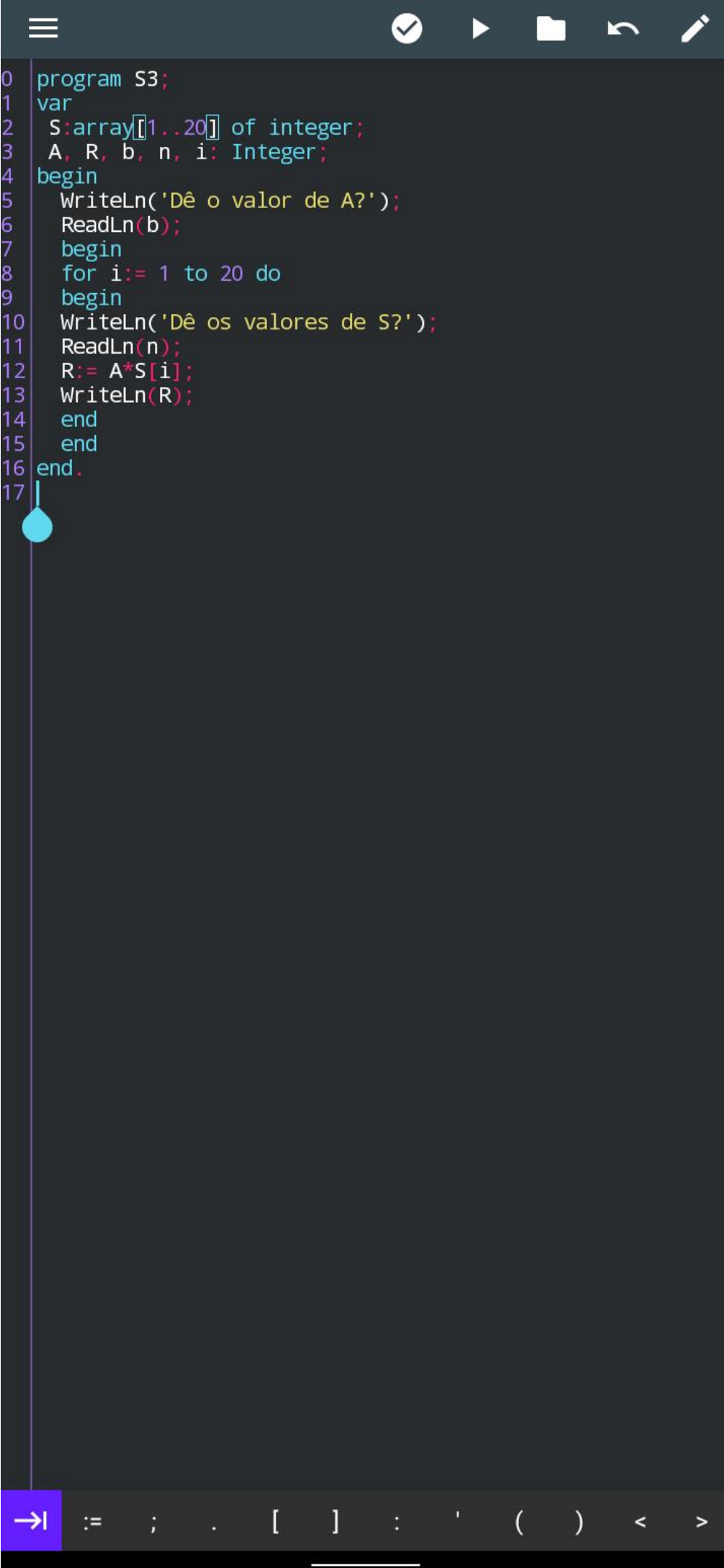
<

>




```
0 program k2;  
1 var  
2   K:array[1..30] of integer;  
3   n, i: Integer;  
4 begin  
5   for i:= 1 to 30 do  
6   begin  
7     Write('Dê os valores de K?');  
8     readln(n);  
9     begin  
10    if n mod 2=1 then  
11    WriteLn(n+1)  
12    end  
13    end  
14 end.  
15
```

 := ; . [ ] : ' ( ) < >



```
0 program S3;
1 var
2   S:array[1..20] of integer;
3   A, R, b, n, i: Integer;
4 begin
5   WriteLn('Dê o valor de A?');
6   ReadLn(b);
7   begin
8     for i:= 1 to 20 do
9       begin
10        WriteLn('Dê os valores de S?');
11        ReadLn(n);
12        R:= A*S[i];
13        WriteLn(R);
14      end
15    end
16 end.
17 |
```



✓

▶

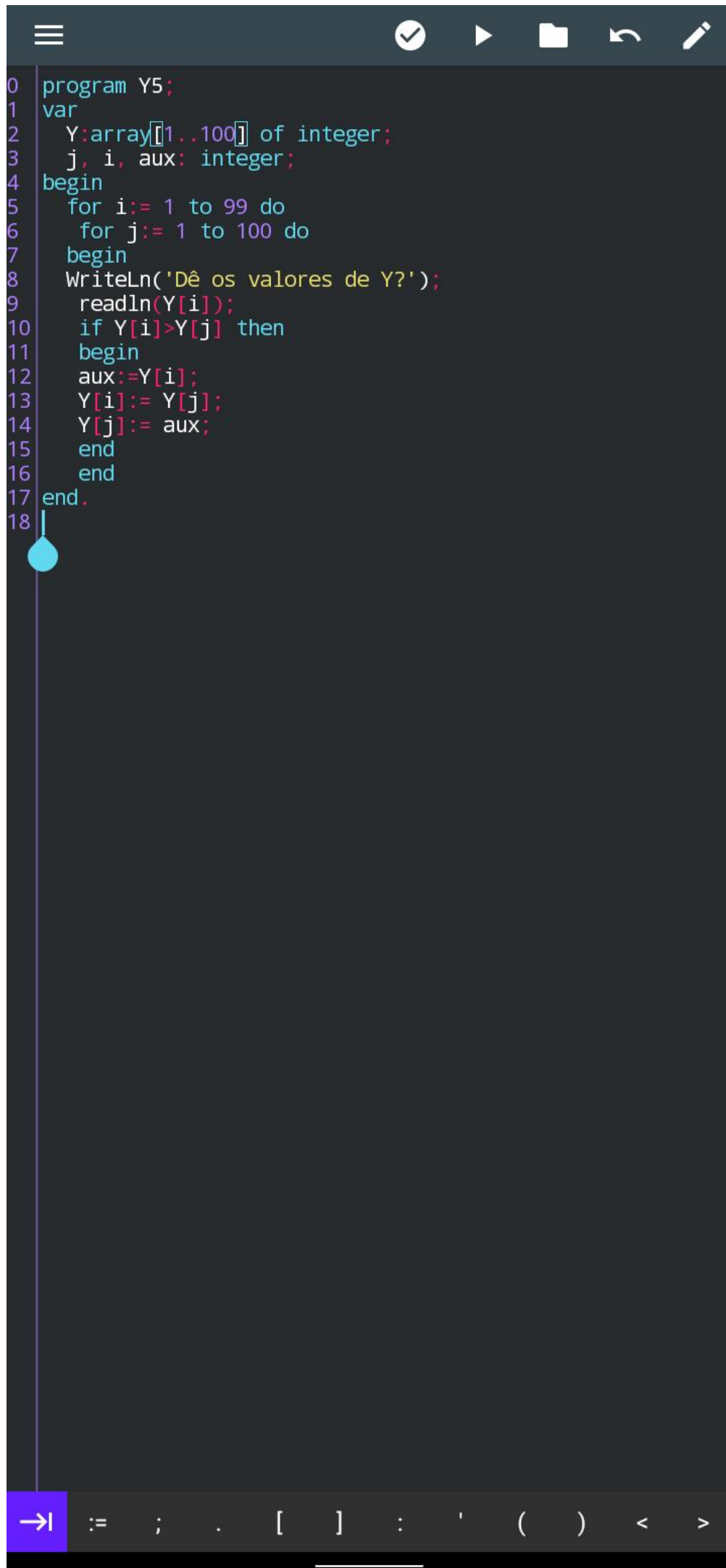
📁

↶

✎

```
0 program L4;
1 var
2   L:array[1..20] of Integer;
3   i, a, n: integer;
4 begin
5   for i:= 1 to 20 do
6     WriteLn('Dê os valores de L?');
7     readln(n);
8     begin
9       if a = mod 2=0 then
10        WriteLn('Existem', a, 'valores pares.')
11      end
12    end.
13
```

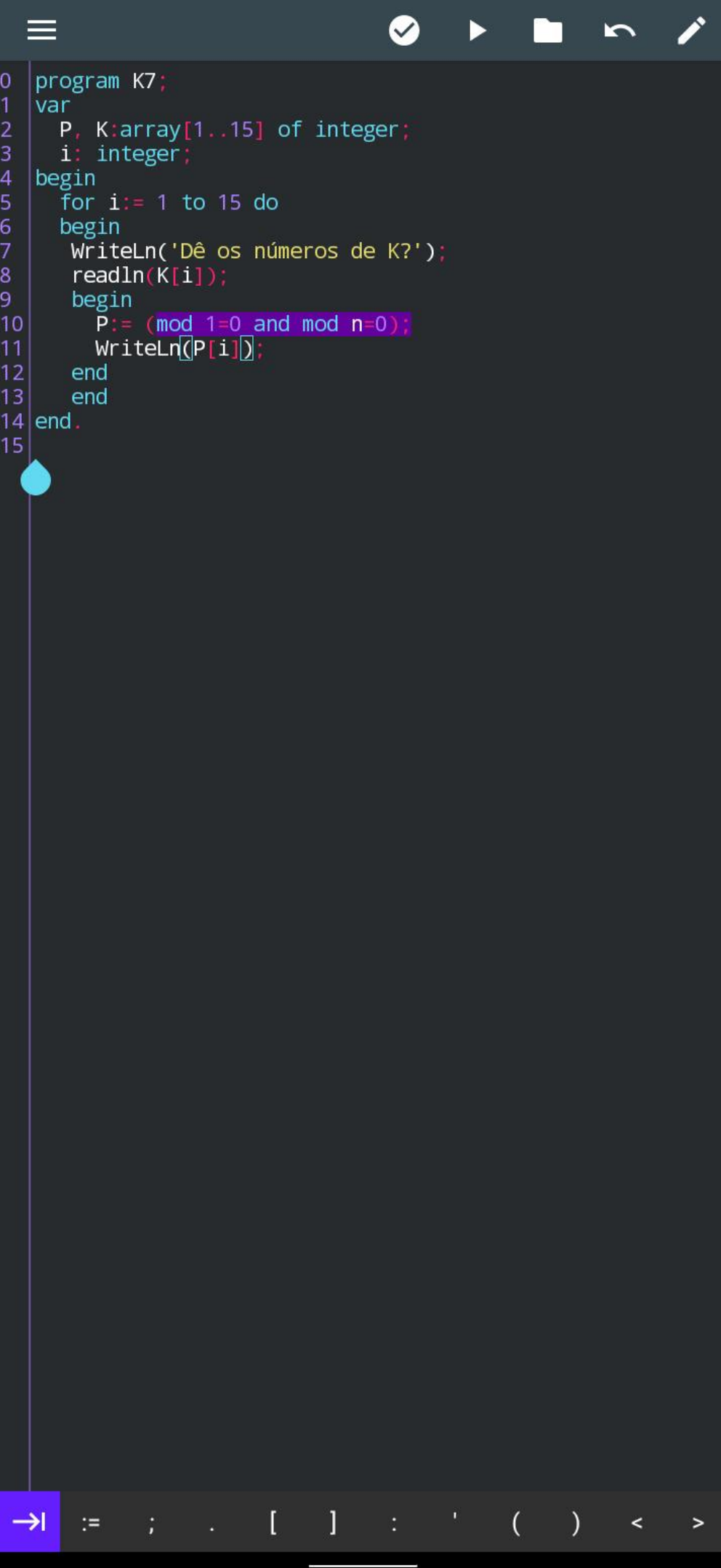
→| := ; . [ ] : ' ( ) < >



The image shows a mobile application interface for a code editor. At the top, there is a dark grey header bar containing a menu icon (three horizontal lines) on the left and a toolbar with five icons: a checkmark, a play button, a folder icon, a circular arrow, and a pencil icon. Below the header is a code editor with a dark background and light-colored text. The code is written in Pascal and is as follows:

```
0 program Y5;  
1 var  
2   Y:array[1..100] of integer;  
3   j, i, aux: integer;  
4 begin  
5   for i:= 1 to 99 do  
6     for j:= 1 to 100 do  
7       begin  
8         WriteLn('Dê os valores de Y?');  
9         readln(Y[i]);  
10        if Y[i]>Y[j] then  
11          begin  
12            aux:=Y[i];  
13            Y[i]:= Y[j];  
14            Y[j]:= aux;  
15          end  
16        end  
17      end.  
18
```

A blue circular cursor is positioned at the end of line 18. At the bottom of the screen, there is a keyboard overlay with a dark grey background. It features a blue button with a right-pointing arrow and a vertical bar, followed by a row of symbols: `:=`, `;`, `.`, `[`, `]`, `:`, `'`, `(`, `)`, `<`, and `>`.



```
0 program K7;  
1 var  
2   P, K:array[1..15] of integer;  
3   i: integer;  
4 begin  
5   for i:= 1 to 15 do  
6   begin  
7     WriteLn('Dê os números de K?');  
8     readln(K[i]);  
9     begin  
10      P:= (mod 1=0 and mod n=0);  
11      WriteLn[P[i]];  
12    end  
13  end  
14 end.  
15
```

→ | = ; . [ ] : ' ( ) < >

0

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

```
program G10;  
var  
  G:array[1..10] of integer;  
  i: Integer;  
begin  
  for i:= 1 to 10 do  
    begin  
      WriteLn('Dê o vetor G?');  
      readln(G[i]);  
      begin  
        if G[i] <=0 then  
          WriteLn(' Os três menores números são', G[i])  
        end  
      end  
    end  
  end.  
end.
```

→|

:

=

;

.

[

]

:

'

(

)

<

>



✓

▶

📁

↶

✎

```
0 program A12;
1 var
2   A:array[1..10] of Integer;
3   i: integer;
4 begin
5   WriteLn('Dê os valores de A?');
6   readln(A[i]);
7   begin
8     aux:=Y[i];
9     Y[i]:= Y[j];
10    Y[j]:= aux;
11    WriteLn(A[i]);
12  end
13 end.
14
```

→|

:= ; . [ ] : ' ( ) < >