

## SIMULADO

**Questão 1:** Leia as informações sobre o comando while e analise o código dos Programas A e B.

```
#include <stdio.h>
int main(){
    int i = 10;
    while (i >= 0){
        printf("%d ", i);
        i = i - 2;
    }
    return 0;
}
```

Assinale a alternativa que corresponde a saída correta do Programa – while.

- A. Imprime os números 0 1 2 3 4 um em cada linha
- B. Imprime os números 1 2 3 todos na mesma linha
- C. Imprime os números 1 2 3 4 5 um em cada linha
- D. Imprime os números 1 2 3 4 todos na mesma linha
- E. Imprime os números 1 2 4 um em cada linha

**Questão 2:** Leia as informações sobre o comando "while" e analise o código abaixo:

```
#include <stdio.h>
int main(){
    int i = 2;
    while (i < 10){
        printf("%d\n", i);
        i = i * 2;
    }
    return 0;
}
```

Assinale a alternativa que corresponde à saída correta do programa:

- A. Imprime os números 2 4 8 todos na mesma linha
- B. Imprime os números 2 4 8 um em cada linha
- C. Imprime os números 2 4 6 um em cada linha
- D. Imprime os números 2 3 4 5 6 em uma linha
- E. Imprime os números 1 3 5 7 9 em uma linha

**Questão 3:** Leia as informações sobre o comando "while" e analise o código abaixo:

```
#include <stdio.h>
int main(){
    int i = 1;
    while (i <= 6){
        printf("%d ", i);
        i = i + 2;
    }
    return 0;
}
```

Assinale a alternativa que corresponde à saída correta do programa:

- A. Imprime os números 1 3 5 em uma linha
- B. Imprime os números 1 3 5 em linhas separadas
- C. Imprime os números 1 3 5 7 9 em uma linha
- D. Imprime os números 1 2 3 4 5 6 em uma linha
- E. Imprime os números 1 3 5 7 em linhas separadas

**Questão 4:** Leia as informações sobre o comando "while" e analise o código abaixo:

```
#include <stdio.h>
int main(){
    int i = 10;
    while (i >= 0){
        printf("%d ", i);
        i = i - 2;
    }
    return 0;
}
```

Assinale a alternativa que corresponde à saída correta do programa:

- A. Imprime os números 10 8 6 4 2 0 em uma linha
- B. Imprime os números 10 8 6 4 2 em uma linha
- C. Imprime os números 10 8 6 4 0 em uma linha
- D. Imprime os números 10 9 8 7 6 em uma linha
- E. Imprime os números 10 8 7 6 5 4 em linhas separadas

**Questão 5:** Leia as informações sobre o comando "while" e o uso de "if-else" e analise o código abaixo:

```
#include <stdio.h>
int main(){
    int i = 1;
    while (i < 10){
        if (i % 2 == 0){
            printf("%d é par\n", i);
        } else {
            printf("%d é ímpar\n", i);
        }
        i = i + 3;
    }
    return 0;
}
```

Assinale a alternativa que corresponde à saída correta do programa:

- A. 1 é ímpar 4 é par 7 é ímpar
- B. 1 é ímpar 4 é par 7 é ímpar 10 é par
- C. 2 é par 5 é ímpar 8 é par
- D. 1 é ímpar 2 é par 3 é ímpar 4 é par
- E. 1 é ímpar 4 é par 6 é par 9 é ímpar

**Questão 6:** Leia as informações sobre o comando "while" e o uso de "if-else" e analise o código abaixo:

```
#include <stdio.h>
int main(){
    int i = 5;
    while (i <= 20){
        if (i % 5 == 0){
            printf("%d é múltiplo de 5\n", i);
        } else {
            printf("%d não é múltiplo de 5\n", i);
        }
        i += 4;
    }
    return 0;
}
```

Assinale a alternativa que corresponde à saída correta do programa:

A.

5 é múltiplo de 5  
9 não é múltiplo de 5  
13 não é múltiplo de 5  
17 não é múltiplo de 5  
20 é múltiplo de 5

B.

5 é múltiplo de 5  
10 é múltiplo de 5  
15 é múltiplo de 5  
20 é múltiplo de 5

C.

5 é múltiplo de 5  
9 não é múltiplo de 5  
13 não é múltiplo de 5  
17 não é múltiplo de 5

D.

6 não é múltiplo de 5  
10 é múltiplo de 5  
14 não é múltiplo de 5  
18 não é múltiplo de 5

E.

5 é múltiplo de 5

10 é múltiplo de 5  
15 é múltiplo de 5  
19 não é múltiplo de 5

**Questão 7:** Leia as informações sobre o comando "while" e o uso de "if-else" e analise o código abaixo:

```
#include <stdio.h>
int main(){
    int i = 0;
    while (i < 6){
        if (i == 3){
            printf("i é igual a 3\n");
        } else {
            printf("i não é igual a 3\n");
        }
        i++;
    }
    return 0;
}
```

Assinale a alternativa que corresponde à saída correta do programa:

A.

i não é igual a 3  
i não é igual a 3  
i é igual a 3  
i não é igual a 4  
i não é igual a 5  
i é igual a 6

B.

i é igual a 3  
i não é igual a 3  
i é igual a 3  
i não é igual a 3  
i não é igual a 3  
i não é igual a 3

C.

i não é igual a 3  
i não é igual a 3  
i é igual a 3  
i não é igual a 4  
i é igual a 5

D.

i é igual a 0  
i é igual a 1  
i é igual a 2  
i é igual a 3  
i é igual a 4  
i é igual a 5

E.

i não é igual a 3  
i não é igual a 3  
i não é igual a 3  
i é igual a 3  
i não é igual a 3  
i não é igual a 3

**Questão 8:** Leia as informações sobre o comando "while" e o uso de "if-else" com os conectivos lógicos "&&" e "||", e analise o código abaixo:

```
#include <stdio.h>
int main(){
    int i = 1;
    while (i < 10){
        if (i % 2 == 0 && i < 5){
            printf("%d é par e menor que 5\n", i);
        } else if (i % 2 == 0 || i == 7){
            printf("%d é par ou é igual a 7\n", i);
        } else {
            printf("%d é ímpar e maior que 5\n", i);
        }
        i += 2;
    }
    return 0;
}
```

Assinale a alternativa que corresponde à saída correta do programa:

A. Imprime:

1 é ímpar e maior que 5  
4 é par e menor que 5  
6 é par ou é igual a 7  
9 é ímpar e maior que 5

B. Imprime:

1 é ímpar e maior que 5  
3 é ímpar e maior que 5  
4 é par e menor que 5  
7 é par ou é igual a 7

9 é ímpar e maior que 5

C. Imprime:

1 é ímpar e maior que 5

3 é par ou é igual a 7

5 é ímpar e maior que 5

6 é par e menor que 5

8 é ímpar e maior que 5

D. Imprime:

1 é ímpar e maior que 5

2 é par e menor que 5

3 é par ou é igual a 7

7 é ímpar e maior que 5

E. Imprime:

1 é ímpar e maior que 5

3 é ímpar e maior que 5

5 é ímpar e maior que 5

7 é par ou é igual a 7

9 é ímpar e maior que 5

**Questão 9:** Leia as informações sobre o comando "while" e o uso de "if-else" com os conectivos lógicos "&&" e "||", e analise o código abaixo:

```
#include <stdio.h>
int main(){
    int i = 3;
    while (i <= 15){
        if (i % 3 == 0 && i > 10){
            printf("%d é múltiplo de 3 e maior que 10\n", i);
        } else if (i % 3 == 0 || i % 5 == 0){
            printf("%d é múltiplo de 3 ou 5\n", i);
        } else {
            printf("%d não é múltiplo de 3 nem de 5\n", i);
        }
        i += 2;
    }
    return 0;
}
```

Assinale a alternativa que corresponde à saída correta do programa:

A. Imprime:

3 é múltiplo de 3 ou 5

5 é múltiplo de 3 ou 5

6 é múltiplo de 3 e maior que 10

7 é múltiplo de 3 e maior que 10  
9 é múltiplo de 3 ou 5  
11 é múltiplo de 3 e maior que 10

B. Imprime:

3 é múltiplo de 3 ou 5  
6 é múltiplo de 3 ou 5  
9 é múltiplo de 3 ou 5  
12 é múltiplo de 3 e maior que 10  
15 é múltiplo de 3 e maior que 10

C. Imprime:

3 é múltiplo de 3 ou 5  
5 é múltiplo de 3 ou 5  
7 não é múltiplo de 3 nem de 5  
9 é múltiplo de 3 ou 5  
12 é múltiplo de 3 e maior que 10  
15 é múltiplo de 3 e maior que 10

D. Imprime:

3 é múltiplo de 3 ou 5  
5 é múltiplo de 3 ou 5  
7 não é múltiplo de 3 nem de 5  
9 é múltiplo de 3 ou 5  
11 não é múltiplo de 3 nem de 5  
13 não é múltiplo de 3 nem de 5  
15 é múltiplo de 3 e maior que 10

E. Imprime:

3 é múltiplo de 3 ou 5  
5 é múltiplo de 3 ou 5  
7 não é múltiplo de 3 nem de 5  
11 é múltiplo de 3 e maior que 10  
15 é múltiplo de 3 e maior que 10



**Questão 10:** Leia as informações sobre o comando "while" e o uso de "if-else" com os conectivos lógicos "&&" e "||", e analise o código abaixo:

```
#include <stdio.h>
int main(){
    int i = 0;
    while (i < 4){
        if (i == 2 || i == 4){
            printf("i é igual a 2 ou 4\n");
        } else if (i != 3 && i < 6){
            printf("i é diferente de 3 e menor que 6\n");
        } else {
            printf("i é maior ou igual a 6\n");
        }
        i++;
    }
    return 0;
}
```

A. Imprime:

i é diferente de 3 e menor que 6  
i é igual a 2 ou 4  
i é igual a 2 ou 4  
i é maior ou igual a 6

B. Imprime:

i é diferente de 3 e menor que 6  
i é diferente de 3 e menor que 6  
i é igual a 2 ou 4  
i é maior ou igual a 6

C. Imprime:

i é diferente de 3 e menor que 6  
i é igual a 2 ou 4  
i é diferente de 3 e menor que 6  
i é igual a 2 ou 4

D. Imprime:

i é igual a 2 ou 4  
i é igual a 2 ou 4  
i é maior ou igual a 6

E. Imprime:

i é diferente de 3 e menor que 6  
i é igual a 2 ou 4  
i é diferente de 3 e menor que 6  
i é maior ou igual a 6  
i é diferente de 3 e menor que 6

**Questão 11:** Analise o código abaixo escrito em C, considere que a variável “tamanho” recebeu do usuário o **valor 10**. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>
int main() {
    int i, tamanho;
    printf("Digite o tamanho do vetor: ");
    scanf("%d", &tamanho);

    int vetor[tamanho];
    for (i = 0; i < tamanho; i +=3) {
        vetor[i] = i;
        printf("%d ", vetor[i]);
    }
    return 0;
}
```

Após a execução do programa será exibida no console:

- A. 1 2 4 6 8
- B. 0 2 4 6 8
- C. 1 2 3 4 5 6 7 8 9 10
- D. 0 1 2 3 4 5 6 7 8 9
- E. o código produz uma repetição infinita

**Questão 12:** Analise o código abaixo escrito em C, considerando que a variável "tamanho" recebeu do usuário o **valor 8**. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>
int main() {
    int i, tamanho;

    printf("Digite o tamanho do vetor: ");
    scanf("%d", &tamanho);

    int vetor[tamanho];

    for (i = 0; i < tamanho; i +=2) {
        vetor[i] = i;
        printf("%d ", vetor[i]);
    }
    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 0 3 6
- B. 1 3 5 7
- C. 0 1 2 3 4 5 6 7
- D. 0 2 4 6
- E. O código produz uma repetição infinita

**Questão 13:** Analise o código abaixo escrito em C, considerando que a variável "tamanho" recebeu do usuário o **valor 8**. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>
int main() {
    int i, tamanho;

    printf("Digite o tamanho do vetor: ");
    scanf("%d", &tamanho);

    int vetor[tamanho];
    for (i = 0; i < tamanho; i++) {
        vetor[i] = i * 2;
        printf("%d ", vetor[i]);
    }
    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 0 2 6 10 14 18 22
- B. 2 6 10 14 18
- C. 0 2 4 6 8 10 12 14
- D. 2 4 6 8 10 12 14
- E. O código não exibe nada no console

**Questão 14:** Analise o código abaixo escrito em C, considerando que a variável "tamanho" recebeu do usuário o **valor 8**. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>
int main() {
    int i, tamanho;

    printf("Digite o tamanho do vetor: ");
    scanf("%d", &tamanho);

    int vetor[tamanho];

    for (i = 0; i < tamanho; i++) {
        if (i % 3 == 0) {
            vetor[i] = i * 2;
        } else {
            vetor[i] = i;
        }
        printf("%d ", vetor[i]);
    }

    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 0 1 2 6 4 5 12 7
- B. 0 1 2 3 4 5 6 7
- C. 0 2 4 6 8 10 12 14
- D. 0 2 4 6 8 12 16
- E. 0 1 2

**Questão 15:** Analise o código abaixo escrito em C, considerando que a variável "tamanho" recebeu do usuário o **valor 5**. Assinale, a seguir, o valor impresso ao final da execução do código.

```

#include <stdio.h>
int main() {
    int i, tamanho;

    printf("Digite o tamanho do vetor: ");
    scanf("%d", &tamanho);

    int vetor[tamanho];

    for (i = 0; i < tamanho; i++) {
        vetor[i] = i++;
        printf("%d ", vetor[i]);
    }

    return 0;
}

```

Após a execução do programa, será exibido no console:

- A. 0 1 2 3 4
- B. 0 2 4
- C. 0 1 1 3 3
- D. 1 2 3 4
- E. O código produz uma repetição infinita

**Questão 16:** Analise o código abaixo escrito em C, considerando que a variável "tamanho" recebeu do usuário o **valor 6**. Assinale, a seguir, o valor impresso ao final da execução do código.

```

#include <stdio.h>
int main() {
    int i, tamanho;

    printf("Digite o tamanho do vetor: ");
    scanf("%d", &tamanho);

    int vetor[tamanho];

    for (i = 0; i < tamanho; i++) {
        vetor[i] = ++i;
        printf("%d ", vetor[i]);
    }

    return 0;
}

```

- A. 1 2 3 4 5 6
- B. 1 3 5
- C. 0 2 4
- D. 1 2 3 4
- E. O código produz uma repetição infinita

**Questão 17:** Analise o código abaixo escrito em C, considerando que a variável "tamanho" recebeu do usuário o **valor 7**. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>
int main() {
    int i, tamanho;

    printf("Digite o tamanho do vetor: ");
    scanf("%d", &tamanho);

    int vetor[tamanho];

    for (i = 0; i < tamanho; i++) {
        vetor[i] = i;
        printf("%d ", ++i);
    }

    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 1 3 5
- B. 0 1 2 3 4 5 6
- C. 1 3 5 7
- D. 1 2 3 4 5 6
- E. O código produz uma repetição infinita

**Questão 18:** Considere os algoritmos 1 e 2, a seguir:

Algoritmo 1	Algoritmo 2	
<pre>#include &lt;stdio.h&gt; int main(){     int x = 0;     if (x &lt;= 4){         x = 5;     }     if (x &gt; 4){         x = x + 4;     }     printf("%d", x);      return 0; }</pre>	<pre>#include &lt;stdio.h&gt; int main(){     int x = 0;     if (x &lt;= 4){         x = 5;     }     else{         x = x + 4;     }     printf("%d", x);      return 0; }</pre>	<p>O resultado final da variável “x”, exibida para o usuário nos Algoritmos 1 e 2, respectivamente é:</p> <p>A. 5 e 0 B. 9 e 9 C. 5 e 5 D. 9 e 5 E. 5 e 9</p>

**Questão 19:** Analise o código abaixo escrito em C. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>
int main() {
    int vetor[] = {4, 9, 3, 12, 5};
    int i;

    for (i = 0; i < 5; i++) {
        if (vetor[i] % 3 == 0) {
            vetor[i] = 0;
        }
    }

    for (i = 4; i >= 0; i--) {
        printf("%d ", vetor[i]);
    }

    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 5 12 0 9 4
- B. 5 0 3 0 4
- C. 5 12 3 9 4
- D. 4 9 0 12 5
- E. 5 0 0 0 4

**Questão 20:** Analise o código abaixo escrito em C. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>
int main() {
    int vetor[] = {6, 2, 8, 1, 7};
    int i;

    for (i = 0; i < 5; i++) {
        if (i % 2 == 0) {
            vetor[i] = vetor[i] + 5;
        }
    }

    for (i = 4; i >= 0; i--) {
        printf("%d ", vetor[i]);
    }
    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 12 1 13 2 11
- B. 7 1 8 2 6
- C. 7 6 13 7 11
- D. 6 2 8 1 7
- E. 12 1 8 2 11



**Questão 21:** Analise o código abaixo escrito em C, considerando que a variável "tamanho" recebeu do usuário o **valor 9**. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>
int main() {
    int i, tamanho;

    printf("Digite o tamanho do vetor: ");
    scanf("%d", &tamanho);

    int vetor[tamanho];

    for (i = 0; i < tamanho; i++) {
        vetor[i] = i * 3;
        printf("%d ", vetor[i]);
    }

    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 1 3 6 9 12 15 18
- B. 0 3 6 9 12 15 18
- C. 1 2 3 4 5 6 7 8 9
- D. 0 2 4 6 8 10
- E. 0 3 6 9 12 15 18 21 24

**Questão 22:** Analise o código abaixo escrito em C, considerando que a variável "tamanho" recebeu do usuário o **valor 8**. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>

int main() {
    int i, tamanho;

    printf("Digite o tamanho do vetor: ");
    scanf("%d", &tamanho);

    int vetor[tamanho];

    for (i = 0; i < tamanho; i++) {
        if (i % 2 == 0 && i != 4) {
            vetor[i] = i * 2;
        } else {
            vetor[i] = i;
        }
        printf("%d ", vetor[i]);
    }

    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 0 1 4 3 4 5 12 7
- B. 0 1 2 3 4 5 6 7
- C. 0 2 4 6 8 10
- D. 0 2 6 8 10 12
- E. O código produz uma repetição infinita

**Questão 23:** Analise o código abaixo escrito em C, considerando que a variável "tamanho" recebeu do usuário o **valor 5**. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>

int main() {
    int i, tamanho;

    printf("Digite o tamanho do vetor: ");
    scanf("%d", &tamanho);

    int vetor[tamanho];

    for (i = 0; i < tamanho; i++) {
        if (i == 3 || i % 2 == 0) {
            vetor[i] = i * 3;
        } else {
            vetor[i] = i + 2;
        }
        printf("%d ", vetor[i]);
    }

    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 0 2 6 9 8
- B. 0 3 6 9 12
- C. 0 1 2 3 4
- D. 1 2 4 6 8
- E. O código não imprime nada

**Questão 24:** Analise o código abaixo escrito em C, considerando que a variável "tamanho" recebeu do usuário o **valor 7**. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>
int main() {
    int i, tamanho;

    printf("Digite o tamanho do vetor: ");
    scanf("%d", &tamanho);

    int vetor[tamanho];

    for (i = 0; i < tamanho; i++) {
        if (i % 3 == 0 && i < 5) {
            vetor[i] = i * 2;
        } else if (i % 3 == 1 || i == 6) {
            vetor[i] = i + 3;
        } else {
            vetor[i] = i;
        }
        printf("%d ", vetor[i]);
    }

    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 0 4 2 6 7 5 9
- B. 0 3 4 6 7 9 11
- C. 1 3 5 7 9 11
- D. 0 4 2 6 8
- E. Nenhuma das alternativas

**Questão 25:** Considere o seguinte código escrito em linguagem de programação C:

```
#include <stdio.h>

int main() {
    int tamanho = 5;

    int vetor[tamanho], contador = 10, i = 0;

    for (i = 0; i < tamanho; i++) {
        vetor[i] = contador--;
    }

    for (i = 0; i < tamanho; i++) {
        printf("%d ", vetor[i]);
    }
    return 0;
}
```

Após a execução do programa será exibida no console:

- A. 1 2 4 6 8
- B. 0 1 2 3 4 5
- C. 1 2 3 4 5 6 7 8
- D. 10 9 8 7 6
- E. 10 9 8 7 6 5

**Questão 26:** Analise o código abaixo escrito em C. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>

int main() {
    int vetor[] = { 10, 15, 20, 25, 30};
    int i;

    for (i = 0; i < 5; i++) {
        if (vetor[i] % 5 == 0 && vetor[i] < 25) {
            vetor[i] = vetor[i] * 2;
        }
    }

    for (i = 4; i >= 0; i--) {
        printf("%d ", vetor[i]);
    }
    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 30 25 20 30 10
- B. 30 25 40 30 20
- C. 30 50 20 25 10
- D. 30 25 40 30 20
- E. 20 25 30 50 10

**Questão 27:** Analise o código abaixo escrito em C. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>
int main() {
    int vetor[] = {9, 6, 3, 12, 15};
    int i;

    for (i = 0; i < 5; i++) {
        if (vetor[i] % 3 == 0) {
            vetor[i] = vetor[i] - 3;
        }
    }

    for (i = 4; i >= 0; i--) {
        printf("%d ", vetor[i]);
    }
    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 12 9 0 3 6 9
- B. 12 9 0 9 12
- C. 9 3 6 9 15
- D. 12 9 0 3 6
- E. 9 12 6 3 15

**Questão 28:** Considere o seguinte código escrito em linguagem de programação C, que preenche e imprime uma matriz:

```
#include <stdio.h>
int main() {
    int matriz[3][3];

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            if (i == j) {
                matriz[i][j] = 1;
            } else {
                matriz[i][j] = 0;
            }
        }
    }

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            printf("%d ", matriz[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

Após a execução do programa, será exibida a seguinte matriz:

A.

```
1 0 0
0 1 0
0 0 1
```

B.

```
1 1 1
0 0 0
1 1 1
```

C.

```
0 0 0
0 1 0
1 1 1
```

D.

```
1 0 1
0 1 0
0 0 1
```

**Questão 29:** Considere o seguinte código escrito em linguagem de programação C, que preenche e imprime uma matriz:

```
#include <stdio.h>
int main() {
    int matriz[3][3];

    // Preenchendo a matriz
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            if (i < j) {
                matriz[i][j] = 1;
            } else {
                matriz[i][j] = 0;
            }
        }
    }

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            printf("%d ", matriz[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

Após a execução do programa, será exibida a seguinte matriz:

A.

```
0 1 1
1 1 1
1 1 1
```

B.

```
0 1 1
0 0 1
0 0 0
```

C.

```
0 1 1
1 1 1
1 0 1
```

D.

```
1 0 0
1 1 0
1 1 1
```



**Questão 30:** Considere o seguinte código escrito em linguagem de programação C, que preenche e imprime uma matriz:

```
#include <stdio.h>
int main() {
    int matriz[3][3];
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            if (i > j) {
                matriz[i][j] = 2;
            } else {
                matriz[i][j] = 3;
            }
        }
    }

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            printf("%d ", matriz[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

Após a execução do programa, será exibida a seguinte matriz:

A.

3 3 3  
2 3 3  
2 2 3

B.

3 3 3  
3 3 3  
3 3 2

C.

2 2 2  
3 3 2  
2 3 3

D.

2 3 2  
3 2 2  
2 3 3

E.

3 3 3

2 3 3

2 2 3

**Questão 31:** Considere o seguinte código escrito em linguagem de programação C, que preenche e imprime uma matriz usando “if-else”:

```
#include <stdio.h>
int main() {
    int matriz[3][3];

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            if (i < j) {
                matriz[i][j] = 1;
            } else if (i == j) {
                matriz[i][j] = 2;
            } else {
                matriz[i][j] = 0;
            }
        }
    }

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            printf("%d ", matriz[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

Após a execução do programa, será exibida a seguinte matriz:

A.

2 0 0

1 2 1

1 1 2

B.

1 1 1

0 0 1

0 0 0

C.

1 2 1

0 1 0

0 1 1

D.

2 1 1  
0 2 1  
0 0 2

**Questão 32:** Considere o seguinte código escrito em linguagem de programação C, que preenche e imprime uma matriz usando “if-else”:

```
#include <stdio.h>
int main() {
    int matriz[4][4];

    for (int i = 0; i < 4; i++) {
        for (int j = 0; j < 4; j++) {
            if (i > j) {
                matriz[i][j] = 3;
            } else if (i == j) {
                matriz[i][j] = 2;
            } else {
                matriz[i][j] = 1;
            }
        }
    }

    for (int i = 0; i < 4; i++) {
        for (int j = 0; j < 4; j++) {
            printf("%d ", matriz[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

Após a execução do programa, será exibida a seguinte matriz:

A.

2 1 1 1  
3 2 1 1  
3 3 2 1  
3 3 3 2

B.

2 2 2 2  
1 1 1 1  
3 3 3 3  
3 3 3 3

C.

1 0 0 0  
0 1 0 0  
0 0 1 0  
1 1 1 1

D.

3 3 3 1  
2 3 1 0  
2 1 0 0  
3 1 1 0

**Questão 33:** Considere o seguinte código C que determina o maior entre três números digitados pelo usuário:

```
#include <stdio.h>

int maiorTresNumeros(int num1, int num2, int num3);

int main() {
    int num1, num2, num3;

    printf("Digite 3 números:\n");
    scanf("%d %d %d", &num1, &num2, &num3);

    printf("O maior é: %d\n", maiorTresNumeros(num1, num2, num3));

    return 0;
}

int maiorTresNumeros(int num1, int num2, int num3) {
    int maior = num1;

    if(num2 > maior)
        maior = num2;
    if(num3 > maior)
        maior = num3;

    return maior;
}
```

Suponha que o usuário digite os seguintes números: 15, 30 e 25. Qual será a saída impressa pelo programa?

- A. O maior é: 15
- B. O maior é: 25
- C. O maior é: 30
- D. O maior é: 45
- E. O maior é: 0

**Questão 34:** Considere o seguinte código escrito em linguagem de programação C, que preenche e imprime uma matriz.

```
#include <stdio.h>

int main() {
    int matriz[3][3], soma = 0;

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            if (i % 2 == 0 && j % 2 == 0) {
                matriz[i][j] = 1;
            } else {
                matriz[i][j] = 0;
            }
            soma = soma + matriz[i][j];
        }
    }

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            printf("%d ", matriz[i][j]);
        }
        printf("\n");
    }

    printf("%d", soma);

    return 0;
}
```

Após a execução do programa será exibida:

A.

1 1 1

1 1 1

1 1 1

Soma = 9

B.

0 0 0

0 0 0

0 0 0

Soma = 0

C.  
1 0 0  
0 1 0  
0 0 1  
Soma = 3

D.  
1 1 1  
0 0 0  
1 1 1  
Soma = 6

E.  
1 0 1  
0 0 0  
1 0 1  
Soma = 4

**Questão 35:** Considere o código abaixo escrito em C. Assinale o valor que será impresso no console ao final da execução do programa.

```
#include <stdio.h>
int multiplicaPorDois(int n) {
    return n * 2;
}

int main() {
    int vetor[] = { 1, 2, 3, 4, 5 };
    int i;

    for (i = 0; i < 5; i++) {
        vetor[i] = multiplicaPorDois(vetor[i]);
    }

    for (i = 0; i < 5; i++) {
        printf("%d ", vetor[i]);
    }
    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 2 3 6 8 10
- B. 1 2 3 4 5
- C. 0 0 0 0 0
- D. 5 4 3 2 1
- E. 2 4 6 8 10

**Questão 36:** Analise o código abaixo escrito em C. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>

int main() {
    int vetor[] = {5, 7, 8, 11, 20};
    int contador = 10, i = 0;

    for (i = 0; i < 5; i++) {
        if (i % 2 == 0)
            vetor[i] = 0;
    }

    for (i = 0; i < 5; i++) {
        printf("%d ", vetor[i]);
    }
    return 0;
}
```

Após a execução do programa será exibida no console:

- A. 5 7 8 11 20
- B. 0 7 8 0 20
- C. 0 7 0 11 0
- D. 20 11 8 7 5
- E. 0 0 0 0 0

**Questão 37:** Considere o seguinte trecho de código em C:

```
#include <stdio.h>

void funcao(int a, int b) {
    int temp = a;
    a=b;
    b = temp;
}

int main() {
    int x = 10, y = 20;
    funcao(x, y);
    printf("x = %d, y = %d\n", x, y);
    return 0;
}
```

Qual será a saída desse programa?

- A. x = 10, y = 20.
- B. x = 20, y = 10.
- C. x = 0, y = 0.
- D. O programa apresentará um erro de compilação.
- E. O programa apresentará um erro de segmentation fault.

**Questão 38:** Considere o seguinte trecho de código em C:

```
#include <stdio.h>

void funcao(int a, int b) {
    int temp = a;
    a = b;
    b = temp;

    printf("x = %d, y = %d\n", a, b);
}

int main() {
    int x = 10, y = 20;

    funcao(x, y);

    return 0;
}
```

Qual será a saída desse programa?

- A. x = 10, y = 20.
- B. x = 20, y = 10.
- C. x = 0, y = 0.
- D. O programa apresentará um erro de compilação.
- E. O programa apresentará um erro de segmentation fault.



**Questão 39.** Considere o seguinte código C que utiliza funções para manipulação de um vetor. Qual será o valor impresso no console ao final?

```
#include <stdio.h>
int soma(int a, int b) {
    return a + b;
}

void modificaVetor(int vetor[], int n) {
    for (int i = 0; i < n; i++) {
        vetor[i] = soma(vetor[i], i);
    }
}

int main() {
    int vetor[] = { 10, 20, 30, 40, 50 };
    int n = 5;

    modificaVetor(vetor, n);

    for (int i = 0; i < n; i++) {
        printf("%d ", vetor[i]);
    }
    return 0;
}
```

Após a execução do programa, será exibido no console:

- A. 10 20 30 40 50
- B. 10 21 32 43 54
- C. 10 22 34 46 58
- D. 11 21 31 41 51
- E. Nenhuma das alternativas anteriores

**Questão 40** A linguagem C foi criada em 1972 nos *Bell Telephone Laboratories* por Dennis Ritchie com a finalidade de permitir a escrita de um sistema operacional. Devido às capacidades e através da divulgação do sistema *Unix* pelas universidades dos Estados Unidos, a linguagem C deixou cedo as portas dos laboratórios Bell, disseminou-se e tornou-se conhecida por todos os tipos de programadores, independentemente dos projetos em que estivessem envolvidos, sendo o livro *The C Programming Language*, de Kernighan & Ritchie, o único elo comum entre os programadores. DAMAS, Luís. **Linguagem C**. 10. ed. São Paulo: Grupo GEN, 2006.

Análise o código abaixo e escolha o bloco de código que deve ser colocado dentro do comando, para imprimir todos os múltiplos de 3 entre 0 e 100.

```

#include <stdio.h>

int main() {
    int i = 0;

    for (i = 0; i <= 100; i++) {
        // bloco de código de uma das alternativas
    }

    return 0;
}

```

**Alternativa:**

<b>A)</b>	<b>B)</b>	<b>C)</b>	<b>D)</b>	<b>E)</b>
<pre> if(i%3 == 0){     printf("%d ",i); } </pre>	<pre> if(i/3 == 0){     printf("%d ",i); } </pre>	<pre> if(i/3 == 0){     printf("%f ",&amp;i); } </pre>	<pre> if(i/3 = 0){     printf("%fd",&amp;i); } </pre>	<pre> if(i%3 = 0){     printf("%fd",&amp;i); } </pre>

**Questão 41:** Considere o código abaixo que utiliza uma função para somar os elementos de um vetor. Qual será o valor impresso ao final da execução do programa?

```

#include <stdio.h>

int somaVetor(int vetor[], int n) {
    int soma = 0;
    for (int i = 0; i < n; i++) {
        soma += vetor[i];
    }
    return soma;
}

int main() {
    int vetor[] = { 1, 2, 3, 4, 5};
    int n = 5;

    printf("Soma: %d\n", somaVetor(vetor, n));
    return 0;
}

```

Após a execução do programa, será exibido no console:

- A. Soma: 10
- B. Soma: 15
- C. Soma: 20
- D. Soma: 5
- E. Soma: 2

**Questão 42:** Um sistema gera um vetor fixo de dados para efetuar o controle de um processo usando o código em C abaixo, em um compilador que trabalha com inteiros de 32 bits:

```
int matriz[] = { 1, 2, 3, 4, 5, 6, 7, 8 };  
int *v, final;  
v = &matriz[1];  
final = v[3];
```

Após a execução desse trecho de código, o valor de final é:

- A. 6
- B. 5
- C. 4
- D. 3
- E. 2

**Questão 43:** Com relação às linguagens de programação C/C++, analise as afirmativas a seguir.

- I. Seja x uma variável do tipo inteiro. Na declaração abaixo, o ponteiro p é inicializado com o endereço de x. `int *p = &x.`
- II. O comando `break` somente pode ser utilizado em conjunto com o comando `switch`.
- III. O comando `return` encerra a execução de uma função.

Está correto o que se afirma em:

- A. I, apenas.
- B. II, apenas.
- C. III, apenas.
- D. I e II, apenas.
- E. I e III, apenas.

**Questão 44:** Considere o seguinte código C que utiliza uma função e ponteiros para trocar os valores de duas variáveis:

```
#include <stdio.h>

void troca(int *a, int *b);

int main() {
    int x = 10, y = 20;

    printf("Antes da troca: x = %d, y = %d\n", x, y);

    troca(&x, &y);

    printf("Depois da troca: x = %d, y = %d\n", x, y);

    return 0;
}

void troca(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}
```

Suponha que o código acima seja executado. Qual será a saída impressa no console?

- A. Antes da troca: x = 10, y = 20  
Depois da troca: x = 10, y = 20
- B. Antes da troca: x = 20, y = 10  
Depois da troca: x = 10, y = 20
- C. Antes da troca: x = 10, y = 20  
Depois da troca: x = 20, y = 10
- D. Antes da troca: x = 20, y = 20  
Depois da troca: x = 10, y = 10
- E. O código resulta em erro de compilação.

**Questão 45:** Analise o código abaixo escrito em C. Assinale, a seguir, o valor impresso ao final da execução do código.

```
#include <stdio.h>

int main() {
    int vetor[] = {5, 7, 8, 11, 20};
    int contador = 10, i = 0;

    for (i = 0; i < 5; i++) {
        if (i % 2 == 0 && i % 2 == 1)
            vetor[i] = 0;
    }

    for (i = 4; i >= 0; i--) {
        printf("%d ", vetor[i]);
    }

    return 0;
}
```

Após a execução do programa será exibida no console:

- A. 5 7 8 11 20
- B. 0 7 8 0 20
- C. 0 7 0 11 0
- D. 20 11 8 7 5
- E. 0 0 0 0 0