

Questão 1



```
1  /*
2   * Arquivo: questao1.c
3   * Compilação: gcc -std=c99 questao1.c -o questao1
4   * Autor: Matheus Souza Da Costa 12021EEL021
5   */
6
7  #include <stdio.h>
8  #include <stdlib.h>
9
10 int main(void) {
11     #if defined(_WIN32) || defined(_WIN64)
12         system("chcp 65001"); // troca para UTF-8
13     #endif
14
15     printf("%d\n", (8>9) && (10!=2)); /* 0 */
16     printf("%d\n", (14 > 100) || (2>1)); /* 1 */
17     printf("%d\n", ( !(14>100) && !(1>2) ) ); /* 1 */
18
19     return EXIT_SUCCESS;
20 }
21
```

Questão 2

```
1  /*
2   * Arquivo: questao2.c
3   * Compilação: gcc -std=c99 questao2.c -o questao2
4   * Autor: Matheus Souza Da Costa 12021EEL021
5   */
6
7  #include <stdio.h>
8  #include <stdlib.h>
9
10 int main(void) {
11     #if defined(_WIN32) || defined(_WIN64)
12         system("chcp 65001"); // troca para UTF-8
13     #endif
14
15     int a;
16
17     scanf("%d", &a);
18     /*o que acontece aqui é que a conta solicitada
19     esta sendo feita mas não existe um comparativo
20     para verificar se esta correto, o correto seria:
21
22     if(a % 2 == 0){
23         printf("O valor é par.\n");
24     }
25
26     Do jeito que se encontra o programa ele sempre
27     irá trazer o print especificado abaixo
28     independente de ser par ou não*/
29     if (a % 2) printf ("O valor é par.\n");
30
31     return EXIT_SUCCESS;
32 }
33
```

### Questão 3

```
1  /*
2  * Arquivo: questao3.c
3  * Compilação: gcc -std=c99 questao3.c -o questao3
4  * Autor: Matheus Souza Da Costa 12021EEL021
5  */
6
7  #include <stdio.h>
8  #include <stdlib.h>
9
10 int main(void) {
11     #if defined(_WIN32) || defined(_WIN64)
12         system("chcp 65001"); // troca para UTF-8
13     #endif
14
15     /* Primeiro que existem muitos erros neste programa
16     sendo que no enunciado não tem o int(main) e também
17     a falta de "{}", então não seria executado nada,
18     entretanto se ele estivesse correto o comando 2 só seria executado se a condição 1 fosse satisfeita.*/
19     if (cond1)
20         if (cond2)
21             comando1;
22     else
23         comando2;
24
25     return EXIT_SUCCESS;
26 }
27
```

### Questão 4

```
1  /*
2  * Arquivo: questao4.c
3  * Compilação: gcc -std=c99 questao4.c -o questao4
4  * Autor: Matheus Souza Da Costa 12021EEL021
5  */
6
7  #include <stdio.h>
8  #include <stdlib.h>
9
10 int main(void) {
11     #if defined(_WIN32) || defined(_WIN64)
12         system("chcp 65001"); // troca para UTF-8
13     #endif
14
15     /*Neste code encontramos um problema meio parecido
16     com o anterior a falta do int(main) e os '{}'
17     mas não considerando esses erros o comando 4 só é executado se a condição 1 e condição 3 não forem satisfeitas*/
18     if (cond1){
19         if (cond2)
20             comando1;
21         else
22             comando2;
23     }
24     else{
25         if(cond3)
26             comando3;
27         else
28             comando4;
29     }
30     return EXIT_SUCCESS;
31 }
32
```

## Questão 5

```
1  /*
2   * Arquivo: questao5.c
3   * Compilação: gcc -std=c99 questao5.c -o questao5
4   * Autor: Matheus Souza Da Costa 12021EEL021
5   */
6
7  #include <stdio.h>
8  #include <stdlib.h>
9
10 int main(void) {
11     #if defined(_WIN32) || defined(_WIN64)
12         system("chcp 65001"); // troca para UTF-8
13     #endif
14
15     int a, rest;
16
17     printf("Digite um número inteiro:");
18
19     scanf("%d", &a);
20
21     rest = a % 2;
22
23     if( rest == 0 && a < 100){
24         printf("O numero é par e menor que 100\n");
25     }
26     else if(rest == 0 && a >= 100){
27         printf("O numero é par e maior ou igual a 100\n");
28     }
29
30     else if(rest != 0 && a < 100){
31         printf("O numero é impar e menor que 100\n");
32     }
33
34     else {
35         printf("O numero é impar e maior que 100\n");
36     }
37
38
39     return EXIT_SUCCESS;
40 }
41
42
```

## Questão 6

```
1  /*
2   * Arquivo: questao6.c
3   * Compilação: gcc -std=c99 questao6.c -o questao6
4   * Autor: Matheus Souza Da Costa 12021EEL021
5   */
6
7  #include <stdio.h>
8  #include <stdlib.h>
9
10 int main(void) {
11     #if defined(_WIN32) || defined(_WIN64)
12         system("chcp 65001"); // troca para UTF-8
13     #endif
14
15     int a, result = 0;
16
17     scanf("%d", &a);
18
19     for (int i = 1; i <= a ; i++) {
20         result += i;
21     }
22     printf("a somatoria dos numeros de 1 até %d é = %d", a, result);
23
24     return EXIT_SUCCESS;
25 }
26
```

## Questão 7

```
1  /*
2   * Arquivo: questao7.c
3   * Compilação: gcc -std=c99 questao7.c -o questao7
4   * Autor: Matheus Souza Da Costa 12021EEL021
5   */
6
7  #include <stdio.h>
8  #include <stdlib.h>
9
10 int main(void) {
11     #if defined(_WIN32) || defined(_WIN64)
12         system("chcp 65001"); // troca para UTF-8
13     #endif
14
15     int i, n;
16
17     scanf("%d", &n);
18     i=1;
19     do{
20         printf("%d\n",i);
21         i = i+1;
22     } while(i <= n);
23     /* se o usuario digitar 0 a saida será 1
24     justamente porque o programa é um contador que
25     inicia em 1 e vai ate n que no caso foi o numero
26     que o usuario colocou*/
27     return EXIT_SUCCESS;
28 }
29
```

## Questão 8

```
1  /*
2   * Arquivo: questao8.c
3   * Compilação: gcc -std=c99 questao8.c -o questao8
4   * Autor: Matheus Souza Da Costa 12021EEL021
5   */
6
7  #include <stdio.h>
8  #include <stdlib.h>
9
10 int main(void) {
11     #if defined(_WIN32) || defined(_WIN64)
12         system("chcp 65001"); // troca para UTF-8
13     #endif
14
15     int n, numero, soma = 0;
16
17     // Solicita ao usuário que insira a quantidade de números (n)
18     do{
19
20         printf("Digite a quantidade de números a serem somados: ");
21
22         scanf("%d", &n);
23
24         // Verifica se n é um número positivo
25
26         if (n <= 0) {
27             printf("Digite um número positivo para a quantidade de números.\n");
28         }
29     } /* se caso o numero não for maior que 0
30     entra em um loop ate que a condição seja satisfeita*/
31     while (n <= 0);
32
33     // Solicita ao usuário que insira os números e calcula a soma
34     printf("Digite os %d números:\n", n);
35     for (int i = 0; i < n; i++) {
36         scanf("%d", &numero);
37         soma += numero;
38     }
39
40     // Imprime a soma dos números
41     printf("A soma dos %d números é: %d\n", n, soma);
42
43     return EXIT_SUCCESS;
44 }
```

## Questão 9

```
1  /*
2  * Arquivo: questao9.c
3  * Compilação: gcc -std=c99 questao9.c -o questao9
4  * Autor: Matheus Souza Da Costa 12021EEL021
5  */
6
7  #include <stdio.h>
8  #include <stdlib.h>
9  #include <string.h>
10 #include <stdbool.h>
11 #include <math.h>
12
13 /*Foi criado uma função afim de colocar o expoente em notação*/
14 char *superscriptinteger(char *buffer, int n){
15     /*aqui temos os expoentes em uma lista que somente funcionará em UTF-8*/
16     /* 0      1      2      3      4 */
17     char superscript[10][4] = {"\u2070", "\u00B9", "\u00B2", "\u00B3", "\u2074",
18                               "\u2075", "\u2076", "\u2077", "\u2078", "\u2079"};
19     /* 5      6      7      8      9 */
20
21     int digit;
22     static bool finished = true;
23
24     if (n == 0 && finished){
25         sprintf(buffer, "%.3s", superscript[0]);
26         return buffer;
27     }
28
29     if (finished) {
30         buffer[0] = '\0';
31         finished = false;
32     }
33
34     /*aqui ele verifica se o expoente será negativo se for o caso*/
35     if (n < 0){
36         strcat(buffer, "\u207B"); // sinal negativo -
37         n = -n;
38     }
39
40     if (n > 0){
41         digit = n % 10;
42         superscriptinteger(buffer, n/10);
43         strcat(buffer, superscript[digit]);
44     }
45
46     if (n == 0){
47         finished = true;
48     }
49
50     return buffer;
51 }
52
53 int main(void) {
54     #if defined(_WIN32) || defined(_WIN64)
55         system("chcp 65001"); // troca para UTF-8
56     #endif
57
58     int n;
59     char expoente[22];
60     // Solicita ao usuário que insira o valor de n
61     do{
62         printf("Digite um inteiro positivo n: ");
63
64         scanf("%d", &n);
65         // Verifica se n é um número positivo
66
67         if (n < 0) {
68             printf("Por favor, insira um número positivo.\n");
69         }
70     } /* se caso o numero não for maior que 0
71     entra em um loop ate que a condição seja satisfeita*/
72     while (n <= 0);
73
74     printf("Potências de 2 de 0 a %d:\n", n);
75
76     // Calcula e imprime as potências de 2
77     for (int i = 0; i <= n; i++) {
78         /*No comentario abaixo não é necessario a biblioteca math.h e não aparece
79         o expoente em notação, e também não será necessaria a mudança para UTF-8*/
80
81         /*unsigned long long int potencia = 1ULL << i; // Calcula 2 elevado à potência i
82         printf("2^%d = %llu reduzido: %e\n", i, potencia, (double)potencia);
83         */
84
85         printf("%d%s = %d => Notação Científica: %e\n", 2, superscriptinteger(expoente, i), (int)pow(2, i), (double)(int)pow(2, i));
86     }
87
88     return EXIT_SUCCESS;
89 }
```



## Questão 10

```
1  /*
2   * Arquivo: questao10.c
3   * Compilação: gcc -std=c99 questao10.c -o questao10
4   * Autor: Matheus Souza Da Costa 12021EEL021
5   */
6
7  #include <stdio.h>
8  #include <stdlib.h>
9
10 int main(void) {
11     #if defined(_WIN32) || defined(_WIN64)
12         system("chcp 65001"); // troca para UTF-8
13     #endif
14
15     int numero;
16     unsigned long long int fatorial = 1;
17
18     do{
19         printf("Entre com um número maior do que 1: ");
20
21         scanf("%d", &numero);
22
23         // Verifica se n é maior que 1
24         if (numero <= 1) {
25             printf("Por favor, insira um número maior que 1.\n");
26         }
27     } /* se caso o numero não for maior que 1
28        entra em um loop ate que a condição seja satisfeita*/
29     while (numero <= 1);
30
31     printf("%d! = ", numero);
32
33     while(numero > 1) {
34         fatorial *= numero;
35         numero--;
36     }
37
38     printf("%llu\n", fatorial);
39
40
41     return EXIT_SUCCESS;
42 }
43
```