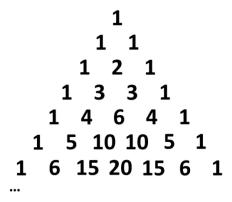
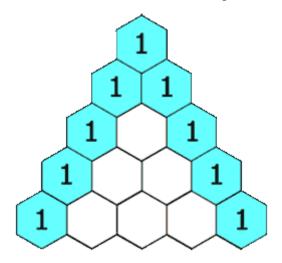
Aluno: João Alexandre Silva do Amaral

**DESAFIOS** 

**Questão 01**. Triângulo de Pascal é um triângulo aritmético infinito onde são dispostos os coeficientes das expansões binominais. Abaixo o exemplo de um Triângulo de Pascal até a linha 7 de altura.



O primeiro e último número de cada linha deve ser 1. Cada número do triângulo de Pascal é igual à soma do número imediatamente acima e do antecessor do número de cima, conforme o gif abaixo demostra.



Agora que você já sabe o que é um triângulo de pascal, faça um programa em C que, dada a altura do triângulo de pascal, desenhe o triângulo resultante.

```
#include <stdio.h>
#define MAX 100
int main() {
  int pascal[MAX][MAX];
  int altura;
```

```
printf("Digite a altura do Triangulo de Pascal: ");
scanf("%d", &altura);
for (int i = 0; i < altura; i++) {
   for (int j = 0; j \le i; j++) {
     if (j == 0 || j == i) {
        pascal[i][j] = 1;
      } else {
         pascal[i][j] = pascal[i-1][j-1] + pascal[i-1][j];
     }
   }
}
for (int i = 0; i < altura; i++) {
   for (int k = 0; k < altura - i - 1; k++) {
      printf(" ");
   }
   for (int j = 0; j \le i; j++) {
      printf("%6d", pascal[i][j]);
   }
   printf("\n");
}
return 0;
```

}

**Questão 02**. Escreva um programa em C que apresente uma pirâmide de números como a que se segue. Para isso deverá pedir ao utilizador o valor de N que deverá ser obrigatoriamente maior ou igual 2. A pirâmide apresentada é o exemplo para N = 5.

|   |   |   | 1 |   |   |   |
|---|---|---|---|---|---|---|
|   |   | 1 | 2 | 1 |   |   |
|   | 1 | 2 | 3 | 2 | 1 |   |
| 1 | 2 | 3 | 4 | 3 | 2 | 1 |

```
1
                      2 3 4 5 4 3 2
#include <stdio.h>
int main() {
  int N;
  do {
     printf("Digite o valor de N (maior ou igual a 2): ");
     scanf("%d", &N);
  \} while (N < 2);
  for (int i = 1; i \le N; i++) {
     for (int s = 0; s < N - i; s++) {
        printf(" ");
     }
     for (int j = 1; j \le i; j++) {
        printf("%2d ", j);
     }
     for (int j = i - 1; j >= 1; j --) {
        printf("%2d ", j);
     }
     printf("\n");
  }
  return 0;
}
```

**Questão 03.** Implemente um programa que desenhe o losango abaixo na tela. O usuário deve informar a largura central do losango.

1

X
XXX
XXXXXX
XXXXXXXX
XXXXXXXX

## **XXXXX**

## XXX

Χ

```
#include <stdio.h>
int main() {
  int largura;
   do {
     printf("Digite a largura central do losango (ímpar e >= 1): ");
     scanf("%d", &largura);
  } while (largura < 1 || largura % 2 == 0);</pre>
   int meio = largura / 2;
  for (int i = 0; i \le meio; i++) {
     for (int j = 0; j < meio - i; j++) {
        printf(" ");
     }
     for (int j = 0; j < 2 * i + 1; j++) {
        printf("X");
     }
     printf("\n");
  }
  for (int i = meio - 1; i \ge 0; i--) {
     for (int j = 0; j < meio - i; j++) {
        printf(" ");
     }
     for (int j = 0; j < 2 * i + 1; j++) {
        printf("X");
     }
     printf("\n");
  }
   return 0;
```

**Questão 04.** Faça um programa em C que apresente um quadrado como o do exemplo abaixo. O usuário deve informar qual a altura ele deseja para o triângulo e se ele gostaria que o triângulo fosse apresentado vazado ou preenchido.

```
*****
******
*****
******
******
******
*****
Preenchido
                                    Vazado
#include <stdio.h>
int main(){
  int altura, vazado;
  printf("\nInforme a altura do quadrado: ");
  scanf("%d", &altura);
  printf("\nVazado --> 0 preenchido --> 1 ");
  scanf("%d", &vazado);
  for(int i = 0; i < altura; i++){
     for(int j = 0; j < altura; j++){
        if(vazado == 1){
        printf(" *");
        }
        if(vazado == 0 \&\& (i == 0 || i == altura - 1 || j == 0 || j == altura - 1)){}
           printf(" *");
        } else {
           printf(" ");
        }
```

```
}
    printf("\n");
}
return 0;
}
```

**Questão 05.** Faça um programa em C que apresente um triângulo como o do exemplo abaixo. O usuário deve informar qual a altura ele deseja para o triângulo e se ele gostaria que o triângulo fosse apresentado vazado ou preenchido.

```
*****
*****
*****
Preenchido
                                    Vazado
#include <stdio.h>
int main() {
  int altura, tipo;
  printf("Digite a altura do triângulo (mínimo 2): ");
  scanf("%d", &altura);
  if (altura < 2) {
     printf("Altura inválida. Deve ser no mínimo 2.\n");
     return 1;
  }
  printf("Escolha o tipo de triângulo:\n");
  printf("1 - Preenchido\n");
  printf("2 - Vazado\n");
  printf("Opção: ");
```

```
scanf("%d", &tipo);
   printf("\n");
   for (int i = 1; i <= altura; i++) {
      for (int j = 1; j <= i; j++) {
         if (tipo == 1) {
            printf("*");
         } else if (tipo == 2) {
            if (j == 1 || j == i || i == altura) {
               printf("*");
            } else {
               printf(" ");
            }
         } else {
            printf("Opção inválida.\n");
            return 1;
         }
      }
      printf("\n");
   }
   return 0;
}
```

Questão 06. Escreva um programa em C que apresente uma árvore de Natal como a do exemplo abaixo. Para isso deverá pedir ao usuário o valor de B que representa o número de asteriscos (\*) na base da árvore. O valor de B deverá ser obrigatoriamente ímpar e maior ou igual 3. Deverá pedir também o valor de L e A que representam, respectivamente a largura e a altura do tronco da árvore. A largura do tronco L deverá ser pelo menos 1 asterisco, deverá ser obrigatoriamente impar e não poderá ultrapassar metade de B, isto é metade do número de asteriscos da base. A altura do tronco A deverá ser pelo menos 2 asteriscos e não deverá ultrapassar metade de B. Em caso de valores inválidos o programa deverá apresentar um erro e voltar a pedi-los.

O exemplo da árvore em baixo tem uma base com B = 13 asteriscos, um tronco com largura L = 3 asteriscos e altura A = 4 asteriscos.

```
#include <stdio.h>
int main() {
  int B, L, A;
   do {
     printf("Digite o valor de B (ímpar e >= 3): ");
     scanf("%d", &B);
     if (B < 3 \parallel B \% 2 == 0) {
        printf("Valor inválido para B. Tente novamente.\n");
     }
  \} while (B < 3 || B % 2 == 0);
   do {
     printf("Digite o valor de L (ímpar, >= 1 e <= %d): ", B / 2);
     scanf("%d", &L);
     if (L < 1 || L \% 2 == 0 || L > B / 2) {
        printf("Valor inválido para L. Tente novamente.\n");
     }
  ) while (L < 1 || L % 2 == 0 || L > B / 2);
   do {
```

printf("Digite o valor de A (>= 2 e <= %d): ", B / 2);

```
scanf("%d", &A);
   if (A < 2 || A > B / 2) {
      printf("Valor inválido para A. Tente novamente.\n");
  }
\} while (A < 2 || A > B / 2);
for (int i = 1; i \le B; i += 2) {
   int espacos = (B - i) / 2;
  for (int j = 0; j < espacos; j++) {
      printf(" ");
   }
   for (int j = 0; j < i; j++) {
      printf("*");
   }
   printf("\n");
}
int espacosTronco = (B - L) / 2;
for (int i = 0; i < A; i++) {
   for (int j = 0; j < espacosTronco; j++) {
      printf(" ");
   }
   for (int j = 0; j < L; j++) {
      printf("*");
   }
   printf("\n");
}
return 0;
```

}