

VV&T Exercício TDD

Nome: Beatriz Lomes

Nusp: 12548038

O primeiro passo a ser feito- primeira versão gerada é a definição da estrutura base com as funções que serão. Fazer esboço do programa

No arquivo .h

```
#ifndef BOLICHE_H
#define BOLICHE_H
```

```
int computaPlacar(const char* jogadas);
```

```
#endif
```

No arquivo .c

```
#include "boliche.h"
```

```
int pontuacaoDoChar(char c) {
    // Implementação será adicionada
}
```

```
int computaPlacar(const char* jogadas) {
    // Implementação será adicionada
}
```

No arquivo test_boliche.c

```
#include <stdio.h>
#include <assert.h>
#include "boliche.h"
```

```
// Função para testar a pontuação do boliche com vários casos
void executeTestCases() {
}
```

```
int main() {
    executeTestCases();
    return 0;
}
```

Na sequência se pensa em um caso mais simples, que seria o calculo sem strikes ou spares, apenas números:

```
#include <stdio.h>
#include "boliche.h" // função de cálculo do placar está definida aqui.
```

```
// Função para testar a pontuação do boliche com vários casos
void executeTestCases() {
    int testCounter = 0;
    int successCounter = 0;

    // Teste simples
    testCounter++;
    if (computaPlacar("111111111111111111") == 20) {
        printf("Teste %d passou.\n", testCounter);
        successCounter++;
    } else {
        printf("Teste %d falhou.\n", testCounter);
    }
}

int main() {
    executeTestCases();
    return 0;
}
```

Depois foi feita a implementação mínima para que pudesse ser testado o caso de teste mais simples proposto:

```
// boliche.c
#include "boliche.h"

int computaPlacar(const char* s) {
    int score = 0;
    for (int i = 0; s[i] != '\0'; i++) {
        score += s[i] - '0'; // Convertendo char para int e somando.
    }
    return score;
}
```

Após esse teste passar, foi proposto outro caso de teste, sendo agora o de strike e spare

```
#include <stdio.h>
#include "boliche.h" // função de cálculo do placar está definida aqui.

// Função para testar a pontuação do boliche com vários casos
void executeTestCases() {
    int testCounter = 0;
    int successCounter = 0;
```

```

// Teste para um jogo só com números
testCounter++;
if (computaPlacar("111111111111111111") == 20) {
    printf("Teste %d passou.\n", testCounter);
    successCounter++;
} else {
    printf("Teste %d falhou.\n", testCounter);
}

// Inclua mais testes aqui, como um jogo com spares, strikes, e combinações

// Teste com um spare
testCounter++;
if (computaPlacar("9-9-9-9-9-9-9-9-9-") == 90) {
    printf("Teste %d passou.\n", testCounter);
    successCounter++;
} else {
    printf("Teste %d falhou.\n", testCounter);
}

// Teste com um strike
testCounter++;
if (computaPlacar("X9-9-9-9-9-9-9-9-9-") == 100) {
    printf("Teste %d passou.\n", testCounter);
    successCounter++;
} else {
    printf("Teste %d falhou.\n", testCounter);
}

// Sumário dos testes
printf("\n%d de %d testes passaram.\n", successCounter, testCounter);
}

int main() {
    executeTestCases();
    return 0;
}

```

E código feito em que esse teste passou foi :

```

// boliche.c
#include "boliche.h"

int pontuacaoDoChar(char c) {
    if (c == 'X' || c == '/') return 10; // Strikes e spares contam como 10 pontos
    if (c == '-') return 0; // Nenhuma pino derrubado
    return c - '0'; // Convertendo char numérico para int
}

```

```

}

int computaPlacar(const char* jogadas) {
    int score = 0;
    int i, frame;

    for (i = 0, frame = 0; frame < 10; ++frame) {
        if (jogadas[i] == 'X') { // Strike
            score += 10 + pontuacaoDoChar(jogadas[i + 1]) + pontuacaoDoChar(jogadas[i + 2]);
            i++; // Pula para a próxima jogada (strikes consomem uma jogada)
        } else if (jogadas[i + 1] == '/') { // Spare
            score += 10 + pontuacaoDoChar(jogadas[i + 2]); // 10 + pontos da primeira bola do
            próximo frame
            i += 2; // Pula para as jogadas do próximo frame
        } else {
            score += pontuacaoDoChar(jogadas[i]) + pontuacaoDoChar(jogadas[i + 1]);
            i += 2; // Pula para as jogadas do próximo frame
        }
    }
    return score;
}

```

Um novo caso de teste foi gerado, agora teste para um jogo perfeito (somente strikes),
#include <stdio.h>
#include "boliche.h" // Sua função de cálculo do placar está definida aqui.

```

// Função para testar a pontuação do boliche com vários casos
void executeTestCases() {
    int testCounter = 0;
    int successCounter = 0;

    // Teste para um jogo só com números
    testCounter++;
    if (computaPlacar("111111111111111111") == 20) {
        printf("Teste %d passou.\n", testCounter);
        successCounter++;
    } else {
        printf("Teste %d falhou.\n", testCounter);
    }

    // Teste com um spare
    testCounter++;
    if (computaPlacar("9-9-9-9-9-9-9-9-9-") == 90) {
        printf("Teste %d passou.\n", testCounter);
        successCounter++;
    } else {
        printf("Teste %d falhou.\n", testCounter);
    }
}

```

```

}

// Teste com um strike
testCounter++;
if (computaPlacar("X9-9-9-9-9-9-9-9-9-") == 100) {
    printf("Teste %d passou.\n", testCounter);
    successCounter++;
} else {
    printf("Teste %d falhou.\n", testCounter);
}

// Teste para um jogo perfeito (somente strikes)
testCounter++;
if (computaPlacar("XXXXXXXXXXXX") == 300) {
    printf("Teste %d passou.\n", testCounter);
    successCounter++;
} else {
    printf("Teste %d falhou.\n", testCounter);
}

// Sumário dos testes
printf("\n%d de %d testes passaram.\n", successCounter, testCounter);
}

int main() {
    executeTestCases();
    return 0;
}

```

E para esse último teste não foi necessário alterar o código da função “computaPlacar(String s)”, que verifica o placar, porque o mesmo já era aprovado.