

Upsolving do Problema A

A aproximação do valor de π pode ser obtida com a resolução da Série de Leibniz:

$$4 \sum_{n=1}^{\infty} \frac{1}{2^n - 1} (-1)^{2n}$$

Assim sendo, a primeira solução **incorreta** foi:

```
#include <stdio.h>

int main(void){
    float e, x = 1.0, y;
    int c = 1;

    while(1){
        y = 1.0 / x;

        if(e >= y){
            break;
        }else{
            x += 2.0;
            c++;
        }
    }

    printf("%d\n", c);

    return 0;
}
```

A inconsistência do algoritmo ocorre na instrução: `if(e >= y){` uma vez que ela é equivalente à `if(y <= e){`. O que contraria a afirmação: “O erro absoluto cometido ao parar no n -ésimo termo é **sempre menor que** o valor absoluto do próximo termo.”.

Portanto, é a solução **correta**:

```
#include <stdio.h>

int main(void){
    float e, t;
    int n = 1;

    scanf("%f", &e);

    while(1){
        t = 1.0 / (2 * n - 1);

        if(t < e){
            break;
        }else{
            n++;
        }
    }

    printf("%d\n", n);

    return 0;
}
```