scanf は C 言語を学ぶ上で誰もがはじめに通る重要な関数で、ユーザーに文字や数値を入力させる時に用います。
しかし、実際の開発現場ではこの scanf 関数を用いることは絶対にありません。
なぜなら、scanf 関数は大変危険な関数だからです。
整数や実数などを扱っているときは特に問題は起きませんが、文字列を扱っている時に問題が発生します。
例えば

```
char s[4];
scanf("%s",s);
```

という処理があったとき、s は文字列の終わりを意味する NULL 文字である'¥0'を除けば半角 3 文字分しか文字を記録することが出来ません。
ところが、scanf 関数には文字数制限が無いので 3 文字以上の文字列の入力を行うことが可能です

この時、「**セグメンテーション違反（segmentation fault)**」という現象が起こります。
これは、アクセスが許可されていないメモリ上のアドレス、もしくは許可されていない方法（読み込み専用のアドレスへ書き込みをしようとしたり、OS の部分を上書きしようとする）でメモリ上の位置にアクセスしようとするときに起こる現象です。セグメンテーション違反が発生するとプログラムは異常終了してしまいます。
つまり、scanf()はこのようなセキュリティ上の問題を抱えているため、開発環境によっては事前に scanf 命令の仕様を禁止しているものも存在します。
代表的な例としてマイクロソフト社の VisualStudio ではバージョン 2017 以降で C/C++のプログラムを作成する際には、**SDL チェック**がデフォルトで有効になっており、これがあるために scanf()関数は通常そのままでは使用不可能になっています。
SDL とは Security Development Lifecycle の略であり、2004 年にマイクロソフト社が提唱した、安全なソフトウェア製品開発のための開発プロセスの事です。
SDL の目的は、ソフトウェアに潜むセキュリティの脆弱性を最小限にし、開発ライフサイクルのできるだけ早い段階で脆弱性を発見して取り除くことにあります。
そのため、セグメンテーション違反を容易に起こす可能性のある scanf()関数の利用は好ましいものではなく、デフォルトの状態ではマイクロソフトの C コンパイラではこの関数を使用禁止にしています。
しかし、scanf()関数がなくなったわけでは無く、利用可能にするにプロジェクトのオプションにある SDL チェックを無効にすれば良いようになっています。
また、セキュリティ上の理由から scanf()の仕様を回避したい場合は、gets 関数を用いたり、VisualStudio2017 移行の場合は scanf_s()関数を用いたりするなどの方法があります。