＜演習課題＞　再帰呼び出し

【課題問題Ⅰ】

正の偶数 n を与え，n 以下の正の偶数の和を求める関数 int sum(int n)を再帰呼び出しを用いて作成したい．以下の式とプログラムの空欄を埋めよ．実行例を右下に示す．

sum(n) = 2+4+6+8+‥‥+n

であるので，

sum(2) = 2

sum(n) = [　　] + n　(n が 2 以外のとき)

と表せる

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

int sum(int n);

int main(void) {
    int value;

    printf("正の偶数を入力して下さい:");
    scanf("%d", &value);
    printf("sum(%d) = %d\n", [　　], [　　]);

    return 0;
}

/* 引数 n は正の偶数 */
int sum(int n) {
    int ret;
    assert(n % 2 == 0 && n > 0);

    if ([　　])
        return 2;
    ret = [　　];

    return ret;
}
```

```
% ./a.out
正の偶数を入力して下さい:2
sum(2) = 2
% ./a.out
正の偶数を入力して下さい:10
sum(10) = 30
```


【課題問題Ⅱ】

以下の仕様を満たすプログラムを作成したい．

仕様：

以下に示すように，文字列を読み込み，その文字列の長さと，順序を逆にした文字列を表示するプログラムを作成せよ．

なお，以下のプロトタイプ宣言により示される関数を作成し，これを用いること．

```
int get_length(char str[LEN]);
```

ここで，関数 get_length は，文字列の格納された配列 str を受け取り，格納されている文字列の長さを返す関数である．また，LEN は，配列の要素数を宣言するために定義されたマクロ名であり，十分大きな値として#define しておくこと．

```
文字列　　：program return
文字列の長さ：7
文字列（逆順）：margorp
```

```
文字列　　：exercise return
文字列の長さ：8
文字列（逆順）：esicrexe
```

上記仕様を満たす完成途中のソースコード /home0/staff/DataStruct/debug/exer2.c をホームディレクトリのどこかにコピーして修正することでプログラムを完成させよ．ただし，次の指示に従い修正すること．

- コンパイル時にエラーがあるときは，エラーメッセージを読み，そのエラーメッセージに対応した修正を施す．出力されたエラーメッセージ毎に，エラーになった理由と具体的にどのように修正したかを表にしてまとめること．
- 実行時に不具合がある場合も同様に不具合の症状とその修正点をまとめて表にしておくこと．

【課題問題Ⅲ】

階乗 n!（n は 0 以上の整数）は，以下のように再帰的に定義される．

$$\begin{cases} n! = n \times (n-1)! \\ 0! = 1, \end{cases}$$

この式を参考にし，再帰呼び出しを用いて 0 以上の整数 n を受け取り $n!$を返す関数 `int Factorial(int n)`を作り，$n!$を求めるプログラムを作成せよ．

また，右の実行例のように関数の最初と最後に再帰の動作がわかる表示を行い，再帰関数の呼び出される順序を分かりやすく表示せよ．

※ヒント：関数 Factorial の中のいろいろな位置に，ロギング（`printf("n=%d\n", n);`）などの一行を追加）して，どのような表示が得られるか試してみよう．

```
% ./a.out
自然数を入力して下さい:5
関数 Factorial(5) にはいりました.
関数 Factorial(4) にはいりました.
関数 Factorial(3) にはいりました.
関数 Factorial(2) にはいりました.
関数 Factorial(1) にはいりました.
関数 Factorial(0) にはいりました.
関数 Factorial(0) から出ます: Factorial(0) = 1.
関数 Factorial(1) から出ます: Factorial(1) = 1.
関数 Factorial(2) から出ます: Factorial(2) = 2.
関数 Factorial(3) から出ます: Factorial(3) = 6.
関数 Factorial(4) から出ます: Factorial(4) = 24.
関数 Factorial(5) から出ます: Factorial(5) = 120.
5 の階乗は，120 です.
```

【課題問題Ⅳ】

ある商品を買うと，次のような仕組みでポイントがたまるとする．

1．初めて購入した時にはポイントが 10 点つく
2．2 個目以降 10 個目までは，1 個購入するたびにポイントは 5 点ずつ加算される
3．11 個目以降は，1 個購入するたびに現在のポイントの 1.1 倍となる
　例：13 個買った場合は 12 個買ったときのポイントの 1.1 倍となる

```
% ./a.out
いくつ買いますか ? : 1
ポイントは 10 点です.
% ./a.out
いくつ買いますか? : 10
ポイントは 55 点です.
% ./a.out
いくつ買いますか ? : 15
ポイントは 86 点です.
```

ただし，ポイントは整数であり，ポイント計算で生じる小数は切り捨てることにする．この商品を一度に n 個購入したときに得られるポイントを求めるプログラムを再帰関数 `int point(int n)` を用いて作成せよ．右上に実行例を示す．