

課題 3-1～3-5 のプログラム例

```
#include <stdio.h>

/* 関数のプロトタイプ宣言 */
void add_num(int *num1, int *num2, int *num3);
void add_absolute(int *num1, int *num2,
                  int *num3);
void divide_date(int date, int *y, int *m,
                 int *d);
void round_value(float v, int *roundup,
                 int *rounddown, int *round);
void to_integer(int *result, char *ch1,
                char *ch2, char *ch3);

/* 仮引数の値の合計を求める */
void add_num(int *num1, int *num2, int *num3)
{
    /* 合計と 0 をそれぞれに代入する */
    *num1 = *num1 + *num2 + *num3;
    *num2 = 0;
    *num3 = 0;
}

/* 仮引数の絶対値の合計を求める */
void add_absolute(int *num1, int *num2,
                  int *num3)
{
    /* 負の場合は-1 をかけて正にする */
    if(*num1 < 0) {
        *num1 *= -1;
    }
    /* 条件演算子を使うこともできる */
    *num2 = *num2 > 0 ? *num2 : -1 * *num2;
    /* 合計を求めると同時に条件演算子を
       使うこともできる */
    *num1 = *num1 + *num2 +
        (*num3 > 0 ? *num3 : -1 * *num3);
    *num2 = 0;
    *num3 = 0;
}

/* 8 桁の整数値を年月日に分割する */
void divide_date(int date, int *y, int *m,
                 int *d)
{
    /* 下 2 桁の値を日とする */
    *d = date % 100;
    /* 下 2 桁を切り捨てる */
    date /= 100;
    /* 同様に月と年も求める */
    *m = date % 100;
    date /= 100;

```

```
        *y = date;
    }

/* 繰り上げた値、切り捨てた値、
   四捨五入した値を求める */
void round_value(float v, int *roundup,
                 int *rounddown, int *round)
{
    /* int にキャストして切り捨てた値を求める */
    *rounddown = (int)v;
    /* 繰り上げた値を求める */
    if(*rounddown < v) {
        *roundup = *rounddown + 1;
    } else {
        *roundup = *rounddown;
    }
    /* 四捨五入した値を求める */
    if((v - *rounddown) >= 0.5) {
        *round = *roundup;
    } else {
        *round = *rounddown;
    }
}

/* 3 つの文字から構成される 3 桁の整数値を求める */
void to_integer(int *result, char *ch1,
                char *ch2, char *ch3)
{
    /* '0' との差を求め変換する */
    *result = (*ch1 - '0') * 100;
    *result = *result + (*ch2 - '0') * 10;
    *result = *result + (*ch3 - '0');
    /* 文字変数を '0' にリセットする */
    *ch1 = '0';
    *ch2 = '0';
    *ch3 = '0';
}

int main(void)
{
    int n1, n2, n3;
    int date1, y1, m1, d1;
    float f1;
    int r1, ru1, rd1;
    int r;
    char c1, c2, c3;
    int cost, v10k, v5k, v1k, vodd;

    /* 課題 3-1 に対する動作確認 */
    n1 = 5; n2 = 7; n3 = -3;
    add_num(&n1, &n2, &n3);
    printf("n1: %d, n2: %d, n3: %d\n",
           n1, n2, n3);

```

```
/* 課題 3-2 に対する動作確認 */
n1 = 6; n2 = -8; n3 = -9;
add_absolute(&n1, &n2, &n3);
printf("n1: %d, n2: %d, n3: %d\n",
       n1, n2, n3);
/* 課題 3-3 に対する動作確認 */
date1 = 20121015;
divide_date(date1, &y1, &m1, &d1);
printf("%d / %d / %d\n", y1, m1, d1);
/* 課題 3-4 に対する動作確認 */
/* 切り捨てと四捨五入が同じ場合 */
f1 = 10.4;
round_value(f1, &ru1, &rd1, &r1);
printf("roundup: %d, rounddown: %d, round: %d\n",
       ru1, rd1, r1);
/* 繰り上げと四捨五入が同じ場合 */
f1 = 10.5;

round_value(f1, &ru1, &rd1, &r1);
printf("roundup: %d, rounddown: %d, round: %d\n",
       ru1, rd1, r1);
/* 全て同じ場合 */
f1 = 10.0;
round_value(f1, &ru1, &rd1, &r1);
printf("roundup: %d, rounddown: %d, round: %d\n",
       ru1, rd1, r1);
/* 課題 3-5 に対する動作確認 */
c1 = '2'; c2 = '6'; c3 = '7';
to_integer(&r, &c1, &c2, &c3);
printf("r: %d, c1: %c, c2: %c, c3: %c\n",
       r, c1, c2, c3);

return 0;
}
```