応用プログラミングＡ　第５回演習問題　クラスの概要(2)　解答例

問題１　仮引数を受け取るコンストラクタ

```
#include <iostream>
#include <cmath>
using namespace std;

class Triangle {
    double a, b, c;
public:
    Triangle(double u, double v, double w);
    void set_a(double x);
    void set_b(double x);
    void set_c(double x);
    double get_a();
    double get_b();
    double get_c();
    double area();
};

Triangle::Triangle(double u, double v, double w) {
    a = u;
    b = v;
    c = w;
}

void Triangle::set_a(double x) {
    a = x;
}

void Triangle::set_b(double x) {
    b = x;
}

void Triangle::set_c(double x) {
    c = x;
}

double Triangle::get_a() {
    return a;
}

double Triangle::get_b() {
    return b;
}

double Triangle::get_c() {
    return c;
}

double Triangle::area() {
    double s;

    s = (a+b+c) / 2;
    return sqrt(s*(s-a)*(s-b)*(s-c));
}

int main() {
    Triangle obj(3.0, 4.0, 5.0);

    cout << "辺a = " << obj.get_a() << "¥n";
    cout << "辺b = " << obj.get_b() << "¥n";
    cout << "辺c = " << obj.get_c() << "¥n";
    cout << "面積= " << obj.area() << "¥n";

    return 0;
}
```

問題２　継承

```
#include <iostream>
using namespace std;

class area_cl2 {
    double height;
    double width;
public:
    void set_h(double h);
    void set_w(double w);
    double get_h();
    double get_w();
};

class rectangle : public area_cl2 {
public:
    rectangle(double h, double w);
    double area();
};

class isosceles : public area_cl2 {
public:
    isosceles(double h, double w);
    double area();
};

void area_cl2::set_h(double h) {
    height = h;
}

void area_cl2::set_w(double w) {
    width = w;
}

double area_cl2::get_h() {
    return height;
}

double area_cl2::get_w() {
    return width;
}

rectangle::rectangle(double h, double w) {
    set_h(h);
    set_w(w);
}

isosceles::isosceles(double h, double w) {
    set_h(h);
    set_w(w);
}

double rectangle::area() {
    return get_h() * get_w();
}

double isosceles::area() {
    return 0.5 * get_h() * get_w();
}

int main() {
    rectangle b(10.0, 5.0);
    isosceles i(4.0, 6.0);

    cout << "長方形高さ:" << b.get_h() << " 幅:" <<
b.get_w() << " 面積:" << b.area() << "¥n";
    cout << "三角形高さ:" << i.get_h() << " 幅:" <<
i.get_w() << " 面積:" << i.area() << "¥n";

    return 0;
}
```

問題３　オブジェクトポインタ

```
#include <iostream>
using namespace std;

class area_cl2 {
    double height;
    double width;
public:
    void set_h(double h);
    void set_w(double w);
    double get_h();
    double get_w();
};

class rectangle : public area_cl2 {
public:
    rectangle(double h, double w);
    double area();
};

class isosceles : public area_cl2 {
public:
    isosceles(double h, double w);
    double area();
};

void area_cl2::set_h(double h) {
    height = h;
}

void area_cl2::set_w(double w) {
    width = w;
}

double area_cl2::get_h() {
    return height;
}

double area_cl2::get_w() {
    return width;
}

rectangle::rectangle(double h, double w) {
    set_h(h);
    set_w(w);
}

isosceles::isosceles(double h, double w) {
    set_h(h);
    set_w(w);
}

double rectangle::area() {
    return get_h() * get_w();
}

double isosceles::area() {
    return 0.5 * get_h() * get_w();
}

int main() {
    rectangle b(10.0, 5.0);
    isosceles i(4.0, 6.0);
    rectangle *pb;
    isosceles *pi;

    pb = &b;
    pi = &i;

    cout << "長方形 高さ:" << pb->get_h() << " 幅:"
<< pb->get_w() << " 面積:" << pb->area() << "¥n";
    cout << "三角形 高さ:" << pi->get_h() << " 幅:"
<< pi->get_w() << " 面積:" << pi->area() << "¥n";

    return 0;
}
```

問題４　共用体

```
#include <iostream>
using namespace std;

union bits {
    bits(int n);
    void show_bits();
    int i;
    unsigned char c[sizeof(int)];
};

bits::bits(int n) {
    i = n;
}

void bits::show_bits() {
    int i, j;
    for (j = sizeof(int)-1; j >= 0; j--) {
        cout << "バイト単位のビットパターン " << j <<
": ";
        for (i = 128; i; i >>= 1) {
            if (i & c[j]) {
                cout << "1";
            } else {
                cout << "0";
            }
        }
        cout << "¥n";
    }
}

int main() {
    int n;

    cout << "整数を入力してください:";
    cin >> n;
    bits ob(n);
    ob.show_bits();

    return 0;
}
```

問題５　すごろくシミュレータ

```
#define NUM 1000
#define GOAL 20

#include <iostream>
#include <cstdlib>
using namespace std;

class PlayerUnit {
    int position;
    int count;
public:
    PlayerUnit() { position = 0; count = 0; }
    void move(int n) { position += n; count++; }
    int getPosition() { return position; }
    int getCount() { return count; }
};

int main() {
    int i, n, sum = 0;

    for (i = 0; i < NUM; i++) {
        PlayerUnit ob;

        while (ob.getPosition() != GOAL) {
            n = rand() % 6 + 1;
            if (ob.getPosition() + n > GOAL) {

ob.move(n-(ob.getPosition()+n-GOAL)  *2);
            } else {
                ob.move(n);
            }
        }
        sum += ob.getCount();
    }
    cout << GOAL << "マス先のゴールにたどり着くのに";
    cout << "平均" << (double)sum/NUM << "回サイコロを振りました¥n";

    return 0;
}
```