

手書き資料のみ持ち込み可

1 動作トレース (25)

```

class Wheel {
private: int size ;
public:
    Wheel( int sz )
        : size( sz ) {}
    int get_size() { return size ; }
};

class Car {
private: char name[ 20 ] ;
public:
    Car( const char n[] ) {
        strcpy( name , n ) ;
    }
    virtual void print() {
        printf( "%s\n" , name ) ;
    }
};

class Engine {
private: double power ;
public:
    Engine( double pw ) : power( pw ) {}
    double get_power() { return power ; }
};

class UniCycle : public Car {
private: Wheel* tire ;
public:
    UniCycle( const char n[] , Wheel* w )
        : Car( n ) , tire( w ) {}
    virtual void print() {
        Car::print() ;
        printf( "-- %d inch\n" ,
            tire->get_size() ) ;
    }
};

class OneWheelBike : public UniCycle {
private: Engine motor ;
public:
    OneWheelBike( const char n[] , Wheel* w , double pw )
        : UniCycle( n , w ) , motor( pw ) {}
    virtual void print() {
        UniCycle::print() ;
        printf( "-- %.f hp\n" , motor.get_power() ) ;
    }
};

void main() {
    Wheel w18( 18 ) ;
    Wheel w20( 20 ) ;
    UniCycle uc( "Kids unicycle" , &w18 ) ;
    OneWheelBike owb( "Bulma's Bike" , &w20 , 100 ) ;
    Car* table[] = {
        &uc , &owb ,
    } ;
    for( int i = 0 ; i < 2 ; i++ )
        table[ i ]->print() ;
}

```

(1) このプログラムの実行結果を答えよ。

(2) このプログラム中の "virtual" を消したプログラムの実行結果を答えよ。

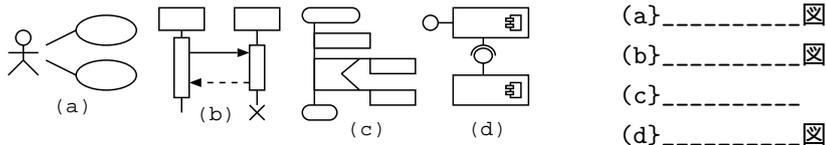
2 説明問題 (25)

以下の用語について説明せよ。

- リファクタリングについて説明せよ。(13)
- is-a , has-a の関係について簡単な事例を交えて説明せよ。(13)

3 用語の穴埋問題 (24)

- オブジェクト指向では、派生クラスが (A) _____ クラスの要素やメソッドを引き継いで利用できることを (B) _____ と呼ぶ。派生クラス毎に (A) クラスの処理に virtual キーワードをつけて再定義することを (C) _____ 関数と呼ぶ。要素や具体的なメソッドが定義されていないクラスから、具体的な要素やメソッドを持つクラスを派生させる手法において元となるクラスは、(D) _____ クラスと呼ぶ。
- 要件定義, 設計, コーディング, テストといった開発のステップを、不具合があっても前工程に戻ることなく開発を行う開発は (E) _____ 型開発と呼ぶ。計画, 実行, チェック, 対策を繰り返すは、(F) _____ サイクルと呼ばれる。これを短期間に繰り返す開発手法は、(G) _____ 開発と呼ぶ。(G) の開発で、プログラムを実装する前にテストコードを書き、それに適合するようにプログラムを書いていく手法は、(H) _____ 開発と呼ばれる。
- 以下の図にふさわしい名称を答えよ。



単語群：アジャイル, ウォーターフォール, オブジェクト, 仮想, ガントチャート, 基底, 継承, コミュニケーション, コンストラクタ, コンポーネント, シーケンス, 状態遷移, タイムチャート, 抽象, テスト駆動開発, デストラクタ, PAD, PDCA, 派生, フローチャート, ペアプログラミング, メソッド, ユースケース, リファクタリング

4 クラス図 (25)

設問1のプログラムにふさわしいクラス図を答えよ。