PB161: PB161_testvnitro_16

Jméno a příjmení – pište do okénka	*UČO*	*Číslo zadání*
		14

Za otázku se všemi správně odpovězenými možnostmi jsou 2 resp. 3 body. Za část správných odpovědí je poměrný počet bodů. Za každou špatnou odpověď je -1 bod. Otázka může mít více správných odpovědí. Není povoleno používat dodatečné materiály.

1

```
#include <iostream>
using namespace std;
class A {
public:
    virtual void foo() const = 0;
    virtual void foo2() const = 0;
};
class B : public A {
public:
    virtual void foo() {}
    virtual void foo2() {}
};
int main() {
    A* var1 = new B;
    return 0;
}
```

Výše uvedený program vypíše při překladu:
..\main.cpp: In function 'int main()':
..\main.cpp:14: error: cannot allocate an object of abstract type 'B'
..\main.cpp:8: note: because the following virtual functions are pure within 'B':
..\main.cpp:5: note: virtual void A::foo() const
..\main.cpp:6: note: virtual void A::foo2() const
Která z uvedených změn umožní kompilaci programu bez chyb?

A Změna metody foo a foo2 ve třídě A z čistě virtuální na metody volané včasnou vazbou
B Přidání modifikátoru const u metody foo a foo2 ve třídě B
C Přetypování vytvářené instance typu B (proměnná var1) na typ A pomocí static_cast
D Odstranění modifikátoru const u metody foo ve třídě A a přidání modifikátoru const u metody foo2 ve třídě B
E Přidání chybějící dealokace dynamicky alokovaného objektu B
F Odstranění modifikátoru const u metody foo a foo2 ve třídě A

2

```
#include <iostream>
int main(){
  int a;
  int* b = &a;
  int*& c = b;

  a = 1;
  *c += 2;
  *b = 2;
  a += 2;
  std::cout<<a<<*b<<*c;
  return 0;
}
```

Pro uvedený kód platí:

A vypíše '322'
B nelze určit, závisí na předešlém obsahu paměti
C vypíše '323'
D žádná z ostatních možností není správná
E vypíše '325'
F vypíše '432'

3

```
#include <iostream>
void print() { std::cout<< "x"; }
namespace MyNamespace {
  void print() {std::cout<< "y";}
}
namespace MyNamespace2 {
  void print() {std::cout<< "z";}
}
int main() {
    print();
    return 0;
}
```

Pro uvedený kód platí:

A nelze přeložit
B vypíše 'z'
C vypíše 'x'
D vypíše 'y'
E vypíše 'yz'
F žádná z ostatních možností není správná

4 Která z uvedených tvrzení jsou pro jazyk C++ pravdivá?

A Operátory nelze přetěžovat, s výjimkou operátoru přiřazení a operátorů pro vstup a výstup
B Většinu operátorů lze přetěžovat, existuje však několik nepřetížitelných operátorů
C Operátor přetěžovaný jako členská metoda má první operand stejného typu, jako třída, která metodu obsahuje.
D Při přetěžování operátorů musí být zachován původní počet operandů
E Lze definovat vlastní symboly operátorů a ty přetěžovat

5
```
#include <iostream>
#include <vector>
class A {
public:
    A() {  std::cout<<"1"; }
    ~A() {  std::cout<<"2"; }
};
int main() {
  std::vector<A*> vect;
  for (int i = 0; i < 3; i++) {
    vect.push_back(new A);
  }
  vect.clear();
  return 0;
}
```
Pro uvedený kód platí:

A vypíše '111222'
B všechna dynamicky alokovaná paměť je korektně uvolněna
C žádná z ostatních možností není správná
D vypíše '111'
E dynamicky alokovaná pamět není korektně uvolněna
F vypíše '222111'

6
```
#include <iostream>
using std::cout;
class X {
  public:
    X() { cout << "X"; }
    virtual ~X() { cout << "~X"; }
};
class Y : public X {
  public:
    Y() { cout << "Y"; }
    Y(const Y& copy) { cout << "cY"; }
    ~Y() { cout << "~Y"; }
};
int main() {
    X   obj1;
    X*   obj2 = new Y;
    delete obj2;
    return 0;
}
```

A žádná z ostatních možností není správná
B vypíše YY~Y~Y
C vypíše řetězec XXY~Y~X~X
D vypíše řetězec XYXY~X~Y~X~Y
E vypíše řetězec YcYYcY~Y~Y
F vypíše řetězec XYXY~Y~X~Y~X

7

```
#include <iostream>
using namespace std;
class A {
public:
    virtual void print() const = 0;
    void print2() const {cout << "7|";};
    virtual ~A() {cout << "4|";}
};
class B : public A {
public:
    virtual void print() const {
        cout << "1|";
    }
    ~B() {cout << "5|";}
};
class C : public A {
public:
    virtual void print() const {
        cout << "2|";
    }
    virtual void print2() const {
        cout << "3|";
    }
    ~C() {cout << "6|";}
};

int main() {
    A* var1 = new B;
    A* var2 = new C;
    var1->print();
    var2->print();
    var2->print2();
    delete var1;
    delete var2;
    return 0;
}
```

Program po spuštění vypíše:

A 1|2|7|4|4|
B program nelze přeložit
C 1|2|7|
D 1|2|3|4|4|
E 1|2|7|5|4|6|4|
F 1|2|3|5|4|6|4|

8

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

class A {
public:
    virtual void foo1() {cout << "A1";}
    void foo2() {cout << "A2";}
    static void foo3() {cout << "A3";}
};

class B : public A {
public:
    virtual void foo1() {cout << "B1";}
    void foo2() {cout << "B2";}
    static void foo3() {cout << "B3";}
};

int main () {
    A* var = new B;

    var->foo1();
    var->foo2();
    var->foo3();

    return 0;
}
```

Uvedený kód vypíše na standardní výstup řetězec:

A A1B2A3
B žádná z ostatních možností není správná
C A1A2B3
D B1B2B3
E B1A2A3
F B1B2A3

Tato strana je prázdná.