**Question 1**

// Assume that integers take 4 bytes.

#include<iostream>

using namespace std;

class Test

{

static int i;

int j;

};

int Test::i;

int main()

{

cout << sizeof(Test);

return 0;

}

Q.2. Write poutput….

#include<iostream>

using namespace std;

class Base1 {

public:

Base1()

{ cout << " Base1's constructor called" << endl; }

};

class Base2 {

public:

Base2()

{ cout << "Base2's constructor called" << endl; }

};

class Derived: public Base1, public Base2 {

public:

Derived()

{ cout << "Derived's constructor called" << endl; }

};

int main()

{

Derived d;

return 0;

}

**Question 3**

#include<iostream>

using namespace std;

class base {

int arr[10];

};

class b1: public base { };

class b2: public base { };

class derived: public b1, public b2 {};

int main(void)

{

cout<<sizeof(derived);

getchar();

return 0;

}

**Question 4**

#include<iostream>

using namespace std;

class P {

public:

void print()

{ cout <<" Inside P::"; }

};

class Q : public P {

public:

void print()

{ cout <<" Inside Q"; }

};

class R: public Q {

};

int main(void)

{

R r;

r.print();

return 0;

}

**Question 5**

#include<iostream>

#include<stdio.h>

using namespace std;

class Base

{

public:

Base()

{

fun(); //note: fun() is virtual

}

virtual void fun()

{

cout<<"\nBase Function";

}

};

class Derived: public Base

{

public:

Derived(){}

virtual void fun()

{

cout<<"\nDerived Function";

}

};

int main()

{

Base\* pBase = new Derived();

delete pBase;

return 0;

}

**Question 6**

#include<iostream>

using namespace std;

class Point {

private:

int x;

int y;

public:

Point(int i = 0, int j = 0); // Normal Constructor

Point(const Point &t); // Copy Constructor

};

Point::Point(int i, int j) {

x = i;

y = j;

cout << "Normal Cunstroctor called\n";

}

Point::Point(const Point &t) {

y = t.y;

cout << "Copy constructor called\n";

}

int main()

{

Point \*t1, \*t2;

t1 = new Point(10, 15);

t2 = new Point(\*t1);

Point t3 = \*t1;

Point t4;

t4 = t3;

return 0;

}

**Question 7**

#include<iostream>

using namespace std;

class Test {

int value;

public:

Test(int v);

};

Test::Test(int v) {

value = v;

}

int main() {

Test t[100];

return 0;

}

**Question 8**

#include<iostream>

using namespace std;

class Test1

{

int x;

public:

void show() { }

};

class Test2

{

int x;

public:

virtual void show() { }

};

int main(void)

{

cout<<sizeof(Test1)<<endl;

cout<<sizeof(Test2)<<endl;

return 0;

}

**Question 9**

#include<iostream>

using namespace std;

class P

{

public:

virtual void show() = 0;

};

class Q : public P {

int x;

};

int main(void)

{

Q q;

return 0;

}

**Question 10**

#include<iostream>

using namespace std;

class A

{

public:

virtual void fun() {cout << "A" << endl ;}

};

class B: public A

{

public:

virtual void fun() {cout << "B" << endl;}

};

class C: public B

{

public:

virtual void fun() {cout << "C" << endl;}

};

int main()

{

A \*a = new C;

A \*b = new B;

a->fun();

b->fun();

return 0;

}

url: http://www.geeksforgeeks.org/c-plus-plus