1. **#include <iostream>**

**using namespace std;**

**int main()**

**{**

**try**

**{**

**throw 'a';**

**}**

**catch (int param)**

**{**

**cout << "int exception\n";**

**}**

**catch (...)**

**{**

**cout << "default exception\n";**

**}**

**cout << "After Exception";**

**return 0;**

**}**

**(A)default exception (B) int exception**

**After Exception After Exception**

**(C) int exception (D) default exception**

1. **#include <iostream>**

**using namespace std;**

**int fun() throw (int)**

**{**

**throw 10;**

**}**

**int main() {**

**fun();**

**return 0;**

**}**

**(A) Compiler error**

**(B) Abnormal program termination**

**(C) Program doesn’t print anything and terminates normally**

**(D) None of the above**

**3. #include <iostream>**

**using namespace std;**

**int main()**

**{**

**int x = ;**

**try {**

**cout << "Inside try n";**

**if (x < 0)**

**{**

**throw x;**

**cout << "After throw n";**

**}**

**}**

**catch (int x ) {**

**cout << "Exception Caught n";**

**}**

**cout << "After catch n";**

**return 0;}**

**(A)Inside try**

**Exception Caught**

**After throw**

**After catch**

**(B)Inside try**

**Exception Caught**

**After catch**

**(C)Inside try**

**Exception Caught**

**(D)Inside try**

**After throw**

**After catch**

**4. #include <iostream>**

**using namespace std;**

**int main() {**

**int x = -1;**

**char \*ptr;**

**ptr = new char[256];**

**try**

**{**

**if (x < 0)**

**{ throw x; } //integer exception**

**if (ptr == NULL)**

**{**

**throw " ptr is NULL "; //string exception**

**}**

**}**

**catch (...)**

**{ cout << "Exception occurred: exiting "<< endl; }**

**return 0;**

**}**

**a) -1**

**b) ptr is NULL**

**c) exception occured: exiting**

**d) 1**

**5. #include <iostream>**

**using namespace std;**

**int main(){**

**int x = -1;**

**try {**

**if (x < 0)**

**{**

**throw x;**

**}**

**else**

**{**

**cout<<x;**

**} }**

**catch (int x )**

**{**

**cout << "Exception occurred: Thrown value is " << x << endl;**

**}**

**return 0;**

**}**

**a) -1**

**b) 0**

**c) Exception occurred: Thrown value is -1**

**d) Error**

**6. #include<iostream>**

**using namespace std;**

**void test(int x) {**

**try {**

**if (x > 0)**

**throw x; //integer exception**

**else**

**throw 'x'; // Character exception**

**} catch (int x) {**

**cout << "Catch a integer and that integer is:" << x <<endl;**

**} catch (char x) {**

**cout << "Catch a character and that character is:" << x<<endl;**

**} }**

**int main() {**

**cout << "Testing multiple catches\n:";**

**test(10);**

**test(0);**

**return 0;}**

**6. #include <iostream>**

**#include <exception>**

**using namespace std;**

**void myunexpected ()**

**{**

**cout << "unexpected called\n";**

**throw 0;**

**}**

**void myfunction () throw (int)**

**{**

**throw 'x';**

**}**

**int main ()**

**{**

**set\_unexpected (myunexpected);**

**try**

**{**

**myfunction();**

**}**

**catch (int)**

**{**

**cout << "caught int\n";**

**}**

**catch (...)**

**{**

**cout << "caught other exception\n";**

**}**

**return 0;**

**}**

**a) caught other exception**

**b) caught int**

**c) unexpected called**

**d) both caught int & unexpected called**

**7. #include <iostream>**

**using namespace std;**

**void empty() throw()**

**{**

**cout << "In empty()";**

**}**

**void with\_type() throw(int)**

**{**

**cout << "Will throw an int";**

**throw(1);**

**}**

**int main()**

**{**

**try**

**{**

**empty();**

**with\_type();**

**}**

**catch (int)**

**{**

**cout << "Caught an int";**

**}**

**}**

**a) In empty()**

**b) Will throw an int**

**c) Caught an int**

**d) All of the mentioned**

**8. #include <iostream>**

**using namespace std;**

**class Test {**

**public:**

**Test()**

**{**

**cout << "Constructing an object of Test " << endl;**

**}**

**~Test()**

**{**

**cout << "Destructing an object of Test " << endl;**

**}**

**};**

**int main() {**

**try {**

**Test t1;**

**throw 10;**

**}**

**catch(int i)**

**{**

**cout << "Caught " << i << endl;**

**}**

**}**

1. **Caught 10**

**(B) Constructing an object of Test**

**Caught 10**

**(C)Constructing an object of Test**

**Destructing an object of Test**

**Caught 10**

**(D) Compiler Error**

**Hint: When an object is created inside a try block, destructor for the object is called before control is transferred to catch block.**

**9. #include <iostream>**

**using namespace std;**

**// Ideally it should have been "int fun() (int)"**

**int fun()**

**{**

**throw 10;**

**}**

**int main()**

**{**

**try**

**{**

**fun();**

**}**

**catch (int )**

**{**

**cout << "Caught";**

**}**

**return 0;**

**}**

**(A) Compiler Error**

**(B) No compiler Error. Output is “Caught”**

**10. Identify the correct statement about throw(type).**

**a) A function can throw any type of exceptions**

**b) A function can throw an exception of certain type only**

**c) A function can’t throw any type of exception**

**d) A function can catch all types of exceptions**