Question 1 (20 marks)

Select the correct option. Every question is worth 1 mark.

1) What is the conventional value to return to the operating system upon the successful completion of a program?

A. -1

B. 1

C. 0

D. Programs do not return a value.

E. true

2) What is the only function all C++ programs must contain?

A. start

B. system

C. main

D. program

E. argc

3) What punctuation is used to signal the beginning and end of a statement blocks?

A. { and }

B. -> and <-

C. BEGIN and END

D. ( and )

E. [ and ]

4) Which of the following are correct comments?

1. \*/ Comments \*/

2. \*\* Comment \*\*

3. /\* Comment \*/

4. { Comment }

5. // Comment

A. 2 and 5

B. All of the above

C. None of the above

D. 1, 3 and 5

E. 3 and 5

5) Which of the following is not a correct variable native type?

A. float

B. real

C. short int

D. double

E. bool

6) Which of the following is the correct operator to compare two variables?

A. :=

B. =

C. equal

D. ==

E. /=

7) What is the value of the local variable x printed to the screen?

int x = 5;

int main()

{

int y = x--;

int x = ++y;

cout << x;

return 0;

}

A. 5

B. 6

C. undefined

D. 1

E. 4

8) Which of the following expressions evaluates to true?

A. 1

B. 15

C. 0.1

D. -1

E. All of the above

9) Which of the following is the Boolean operator for logical-and?

A. &

B. &&

C. |

D. |&

E. ^

10) Evaluate the expression !(0 && !(0 || (3 == 2+1))).

A. true

B. false

C. The expression is ill-formed and cannot be evaluated

D. 4

E. 0

11) What is the final value of x when after the loop terminates?

int x;

for(x=0; x<10; x++) {}

A. 10

B. 9

C. 11

D. 0

E. x no longer exist

12) When does the statement following while(x<100) execute?

A. When x is less than one hundred

B. When x is greater than one hundred

C. When x is less or equal to one hundred

D. While x is equal to one hundred

E. Never

13) What is the final value of x after the loop terminates?

for(signed char x = 0; x < 128; x++) {}

A. 127

B. 128

C. -128

D. -127

E. the loop never terminates

14) Which is not a valid function head?

A. int foo(char x, char y);

B. double foo(char x);

C. void foo();

D. char x();

E. foo(int x);

15) given the function with function head 'void foo()', which of the following is a valid function call?

A. foo;

B. foo x, y;

C. foo();

D. void foo();

E. (\*foo)();

16) Which of the following are syntactically correct and complete (i.e. having both head and body) functions?

1. int foo();

2. int foo(int x) {return x=x+1;}

3. void foo(int) {cout<<"Hello";}

4. void foo(x) {cout<<"Hello"}

5. void (\*foo) {cout<<"Hello";}

A. 1, 2, 3, and 4

B. All of the above

C. 1

D. 2 and 3

E. 2, 3 and 4

17) Which of the following is/are the proper declaration(s) of a pointer?

1. int x;

2. int &x;

3. ptr x;

4. int \*x;

5. double \*\*x;

A. 4, and 5

B. All of the above

C. 2

D. 3

E. None of the above

18) What is the value of x after this code is run?

int a = { 10, 20, 30, 40 };

int x = (a+2)[1];

A. 10

B. 20

C. 30

D. 40

E. The code is incorrect

19) What keyword covers unhandled possibilities in a switch statement?

A. all

B. contingency

C. default

D. other

E. any

20) What is the correct type to declare a variable f as a function pointer to a function 'int foo(double)'?

A. int foo(double) f;

B. int (f\*)(double);

C. int (\*f)(double);

D. int (double) \*f;

E. none of the above;

Question 2 (20 marks)

Although omitted, assume that all following small programs starts with the two lines of code:

#include <iostream>

using namespace std;

Each of the following small programs has one error which the compiler can detect, therefore it either will not compile or will compile with warning but fail to link. Find the error and explain it.

1. 5 marks

struct A

{

A() { x = foo(); }

virtual int foo() const = 0;

int x;

};

struct B : public A { virtual int foo() const { return 5; } };

int main()

{

B b;

cout << b.x << endl;

return 0;

}

1. 5 marks constructor void -wrong

struct C

{

void C() : x(0) {}

const int operator() const { return 2+x; }

int x;

};

int main()

{

C b;

cout << b.x << b() << endl;

return 0;

}

1. 5 marks

struct A

{

A() : x(0) {}

virtual int foo() const = 0;

int x;

};

struct B : public A {};

int main()

{

B b;

cout << b.x << endl;

return 0;

}

1. 5 marks

int& foo(int x)

{

return 10 + x;

}

Question 3 (20 marks)

Although omitted, assume that all following small programs have the correct ***include*** and ***using*** statements (e.g. ***include <iostream>*** and ***using namespace std***.).

Each of the following small programs has one error which the compiler cannot detect, but that will cause problems at runtime. Find the error and explain it.

1. 5 marks a is stack not heap, so needn’t delete!!!

int foo()

{

int a[5] = {1};

int \*p = a+2;

\*(++p) = 4;

for (int i = 0; i < 5; ++i)

cout << a[i] << endl;

delete p;

return 0;

}

1. 5 marks mark

struct A { A() {} };

struct B : private A

{

B(int n) : v(n) {}

vector<double> v;

};

void foo()

{

const A \*a = new B(5);

delete a;

}

1. 5 marks

unsigned sum(unsigned n)

{

unsigned s=0;

while (--n >= 0)

s += n;

return s;

}

1. 5 marks

vector<double> sum(unsigned n)

{

vector<double> v;

for (int j = 0; j < n; ++j)

v[j] = 2\*j;

return v;

}

Question 4 (8 marks)

Trace step by step all local variables (n, k, i, nu, de) and the output of the following program:

#include <iostream>

#include <algorithm>

using namespace std;

int main()

{

unsigned n = 4;

unsigned k = 2;

unsigned nu = 1, de = 1;

for (unsigned i = 1; i <= k; ++i) {

de \*= i \* (n - i) + 1;

nu \*= (n-i+1);

}

cout << nu << " " << de << " " << (nu / de) << "\n";

return 0;

}

Question 5 (12 marks)

Given a vector

vector<pair<int,int>> v;

implement a functor predicate which can be passed to the function std::sort to sort the vector in descending order based on the value of the second element of the pair.

Question 6 (20 marks)

1. [15 marks] Implement a function which computes the price of a European derivative of a non dividend stock price S observed at time T with payoff *f*(ST), via Monte Carlo simulation. Optimize as much as possible computations (i.e. avoid unnecessary and duplicated calculations).

The function has the following function head:

template <class F>

( const F& f // payoff function implemented as a functor

, unsigned n // number of paths

, double S // spot price of the stock (at time t=0)

, double r // risk free rate

, double v // volatility of the stricken price

, double T // time to maturity

);

The formula to implement is:

where Zi are independent random numbers with distribution N(0,1).

Gaussian random numbers can be obtained with the following algorithm, which, given two random numbers U1 and U2 with continuous uniform distribution in the interval [0,1], generates two random numbers Z1 and Z2 with standard Gaussian distribution N(0,1)

The function std::rand() generates integer random numbers with uniform discrete distribution in the interval [0, RAND\_MAX]. Random numbers with continuous uniform distribution in the interval [0,1] can be approximated dividing them by RAND\_MAX.

1. [3 marks]

Implement the payoff for a European call option as a functor, where the strike is a data member of the functor initialized in the constructor.

1. [2 marks]

Show how to call the function constructed in part (a) and the functor defined in part (b) to price a European call option with strike 5, maturity T=2, on a non-dividend paying stock with Spot=4.9, volatility V=20%, when the interest rate is 5%, using 10,000 simulations.