

Assume that all variables without declaration shown in the following program segments have already been declared properly. Integers in problem statements are 32-bit signed variables (**Pascal**: `longint`, **C**: `int`). Assume all the programs are compiled properly without using any compiler flag (except the `-o` option in C).

	Format	# Questions	Total Marks
<b>Section A</b>	Multiple Choice	25	25
<b>Section B</b>	Fill-in-the-blanks	6 (A-O)	20
<b>Total</b>			<b>45</b>

### Section A (25 marks)

For each question, choose the most appropriate answer and write the letter (**A**, **B**, **C** or **D**) in the corresponding space on the answer sheet. One mark for each correct answer. No marks will be deducted for wrong answers.

1. Which of the following statements is/are true?
  - i. The ASCII code of 'A' is smaller than that of 'a'.
  - ii. The ASCII code of ' ' (space) is smaller than that of '0'.
  - A. None of the statements
  - B. i only
  - C. ii only
  - D. i and ii
  
2. Assume we have a box with infinite number of socks inside, and we can take a sock of random color from the box each time. Which of the following statements must be true?
  - i. If only blue socks and red socks are put into the box, to guarantee that we take at least one pair of socks of same color, we need to take out at least 3 socks blindly from the box.
  - ii. If there are  $N$  colors of socks inside the box, to guarantee that we take at least one pair of socks of same color, we need to take out at least  $\frac{1}{2}N(N + 1)$  socks blindly from the box.
  - iii. If there are  $N$  colors of socks inside the box, to guarantee that we take  $K$  socks of same color, we need to take out at least  $\frac{1}{2}N(N + 1)(K - 1)$  socks blindly from the box

Cancelled.

3. Assume the program segment below takes exactly 2 seconds to run on a certain computer.

**Pascal Version**

```
sum := 0;
for i := 1 to 10000 do
begin
    sum := sum+i;
end;
```

**C Version**

```
sum = 0;
for (i = 1; i <= 10000; i++) {
    sum = sum + i;
}
```

Now, consider the following program:

**Pascal Version**

```
sum := 0;
for i := 1 to 20000 do
begin
    sum := sum+i;
    if (i > 0) then
    begin
        sum := sum-i;
        i := i+1;
    end;
end;
```

**C Version**

```
sum = 0;
for (i = 1; i <= 20000; i++) {
    sum = sum + i;
    if (i > 0) {
        sum = sum - i;
        i++;
    }
}
```

Which of the following best approximates the time required to run the above program segment on the same computer?

- A. 2 seconds
- B. 4 seconds
- C. 6 seconds
- D. 8 seconds

4. **Cancelled. The original question was missing information.**

In a village, there are  $N$  ( $N \geq 4$ ) houses that were not yet connected to each other. Now the elder of the village decides to build roads between houses so as to make travelling between houses more convenient. However, due to a mysterious tradition, three houses cannot be directly connected by roads. (That means, if there are such combination of house A, B, C such that there are roads between A and B, B and C, and C and A, bad things will happen to the village). *The roads must not cross each other.*

At most how many roads can be built while following the ritual?

- A.  $N$
- B.  $2N - 4$
- C.  $\frac{1}{2}(N - 1)(N - 2)$
- D. The number of roads are different for all  $N$  and no fixed formula can be found

5. In a mysterious language, some words are regarded as wonderful words. The first and the last letter of a wonderful word must be 'A' while the rest are 'B'. Moreover, the wonderful word with  $x$  'B's is called the  $x$ -wonderful word. For example, "ABA" is the 1-wonderful word. "ABBA" is the 2-wonderful word. "ABBBA" is the 3-wonderful word, etc.

Now we introduce wonderful sentences. The 1-wonderful sentence is the 1-wonderful word, "ABA". The 2-wonderful sentence is constructed by inserting a 2-wonderful word between two 1-wonderful sentence. It is "ABAABBAABA". And the 3-wonderful sentence is constructed by inserting a 3-wonderful word between two 2-wonderful sentence, etc. If this process continues infinitely, we can form an infinitely-wonderful sentence. The infinitely-wonderful sentence starts like this:

ABAABBAABAABBBAABAABBAABAABBBBA...

The first letter is "A" and the second letter is "B". Which of the following statements are true?

- i. The 50th letter is "B".
- ii. The 55th letter is "B".
- iii. The 60th letter is "A".
- iv. The 70th letter is "A".

- A. i and ii
- B. i and iii
- C. ii and iv
- D. iii and iv

6. Long long ago, villagers use four kinds of object (rice, cabbage, pork and water) for bartering (goods trading).

36 kilograms of rice can be exchanged for 17 kilograms of cabbage and vice versa.

13 kilograms of pork can be exchanged for 18 kilograms of rice and vice versa.

12 gallons of water can be exchanged for 49 kilograms of pork and vice versa.

(Note: You can trade in non-integral amounts, e.g. exchange 18 kilograms of rice for 8.5 kilograms of cabbage)

Which of the following has the highest value?

- A. 101 kilograms of rice
- B. 48 kilograms of cabbage
- C. 74 kilograms of pork
- D. 18 gallons of water

7. What is the output of the following program?

#### Pascal Version

```

var c:longint;
function f(a,b:longint):longint;
var tmp:longint;
begin
  tmp := 0;
  if (a = 1) then
    tmp := 1
  else if (b = 1) then
    tmp := 1
  else
    tmp := f(a-1, b) + f(a, b-1);
  f := tmp;
end;
begin
  c := f(2, 3) + f(3, 3);
  write(c);
end.

```

#### C Version

```

#include <stdio.h>
int c;
int f(int a, int b){
  int tmp = 0;
  if (a == 1) {
    tmp = 1;
  } else if (b == 1) {
    tmp = 1;
  } else {
    tmp = f(a-1, b) + f(a, b-1);
  }
  return tmp;
}
int main() {
  c = f(2, 3) + f(3, 3);
  printf("%d", c);
  return 0;
}

```

- A. 5
- B. 7
- C. 9
- D. 11

8. Which of the following statements about data structures may be wrong?

- A. Binary search can be applied to an array
- B. Bubble sort can be applied to a linked list
- C. Merge sort can be applied to a stack
- D. Linear search can be applied to a queue

9. How many ways, disregarding order, are there to choose any three different prime numbers between 4 and 20 inclusive?

- A. 20
- B. 70
- C. 120
- D. 170

10. The following is the truth table for operator  $\diamond$ :

A	B	$A \diamond B$
True	True	True
True	False	True
False	True	True
False	False	False

Simplify the following boolean expression, where U and V are boolean variables

$$(\text{NOT } (U) \diamond (U \diamond V)) \text{ AND } V$$

- A. True
- B. False
- C. U
- D. V

11. There are many people in a party. Some pairs of people are friends. Their relationship is interesting. If A and B are friends, and B and C are friends, then A and C would be friends too. You are given the following information about the people in the party:

Tryndamere and Taric are friends.  
 Master Yi and Wukong are friends.  
 Xin Zhao and Jarvan IV are friends.  
 Lee Sin and Garen are friends.  
 Taric and Xin Zhao are friends.

At least how many pairs of friends are there?

- A. 5
- B. 6
- C. 7
- D. 8

12. What is the name of the sorting algorithm used in the program below?

**Pascal Version**

```
var
  a:array[0..4] of longint;
  i, j, k:longint;
begin
  read(a[0], a[1], a[2], a[3], a[4]);
  for i := 0 to 4 do
    begin
      j := a[i];
      k := i-1;
      while (k >= 0) do
        begin
          if (a[k] < j) then break;
          a[k+1] := a[k];
          k := k-1;
        end;
      a[k+1] := j;
    end
  end.
```

**C Version**

```
#include <stdio.h>
int a[5];
int i, j, k;
int main() {
  scanf("%d%d%d%d%d", &a[0], &a[1], &a[2],
    &a[3], &a[4]);
  for (i = 0; i < 5; i++) {
    j = a[i];
    k = i-1;
    while (k >= 0) {
      if (a[k] < j) break;
      a[k+1] = a[k];
      k = k-1;
    }
    a[k+1] = j;
  }
  return 0;
}
```

- A. Regular sort
- B. Bubble sort
- C. Insertion sort
- D. Selection sort

13. What is the output of the following program segment?

**Pascal Version**

```
var x:real;
...
x := 0.0000;
while (x <> 42.0000) do
begin
  x := x+0.1000
end;
write(x:0:4);
```

**C Version**

```
double x = 0.0000;

while (x != 42.0000) {
  x = x+0.1000;
}
printf("%.4f", x);
```

- A. 43.0000
- B. 42.0000
- C. 41.9999
- D. None of the above

14. Consider the following program segment. Variable  $y$  is an integer while  $x$  is declared as a certain data type and contains integer value in  $[-200, 200]$ .

**Pascal Version**

```
y := -1;
if (x < 0) then
  y := 0
else if (x < 100) then
  y := 1
else if (x >= 100) then
  y := 2;
write(y);
```

**C Version**

```
y = -1;
if (x < 0) {
  y = 0;
} else if (x < 100) {
  y = 1;
} else if (x >= 100) {
  y = 2;
}
printf("%d", y);
```

Which of the following statements must be true?

- A. If  $x$  is declared as double (**Pascal:** real), the program may output -1.
- B. If  $x$  is declared as int (**Pascal:** longint), the program may output -1.
- C. If the value of  $x$  is negative, the program may output something other than 0.
- D. The outputs are always the same no matter  $x$  is declared as integer or double (**Pascal:** real) for the same value  $x$ .

15. What is the output of the following program segment?

**Pascal Version**

```
write(12 and 4, ' ', 10 or 21);
```

**C Version**

```
printf("%d %d", 12 & 4, 10 | 21);
```

- A. 4 0
- B. 4 31
- C. 12 1
- D. 12 10

16. Which of the following data types can be used as the control variable in a switch (case-of) statement?

- i. char
- ii. double (**Pascal:** real)

- A. None of the two types
- B. i only
- C. ii only
- D. i and ii

17. David, a very intelligent student, liked cookies very much. Everyone wanted to give their cookies to David in order to receive help for passing the exam. Therefore David said: "If you have the largest number of cookies among your classmates, I would offer private tutorial class."

Suppose Alice and Bob got the most amount of cookies in class A and B respectively, and the sum of their owned cookies equaled  $S$ . Which of the following statements must be true?

- i. Alice and Bob would both pass the exam.
- ii.  $S$  was greater than the sum of the cookies owned by classmates of Alice and Bob.
- iii. Picking any one student from class A and any one student from class B respectively, the sum of their owned cookies would be smaller than  $S$ .
- iv. For any student in class A, the number of cookies the student owned would be smaller than the difference between  $S$  and the cookies owned by Bob.

- A. None of the statements
- B. i only
- C. ii only
- D. iii and iv only

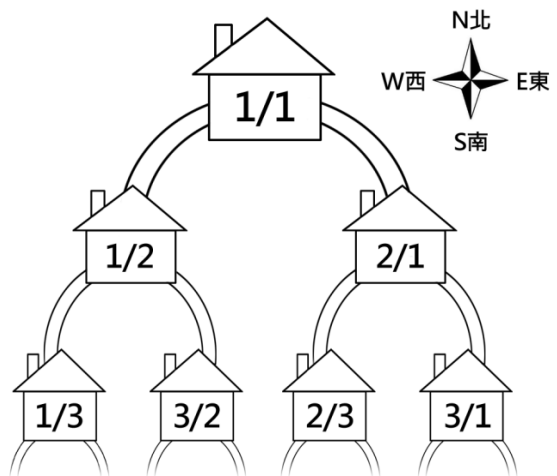
18. There are infinite amount of \$0.5, \$1 and \$2 coins. You can pick the coins one by one until you have collected exactly \$4.5. How many different ways of picking are there? (The order matters)

Cancelled.

19. There is a indefinitely large village. Each house, which named as a rational number  $p/q$ , leads to two other houses to the south. The house to the west is  $p/(p+q)$  while the house to the east is  $(p+q)/q$ . The house names (rational numbers) are never simplified. Since the north-most house is named  $1/1$ , the map of the village looks like this:

By walking only to the south from house  $1/1$ , which of the following houses can you reach by going through 5 other houses? For example, we can reach the house  $3/2$  if we going through 1 house, which is  $1/2$ .

- A.  $17/9$
- B.  $17/10$
- C.  $13/15$
- D.  $11/8$





20. What is the output of the following program?

**Pascal Version**

```
var
  i, j:longint;
  a:array[0..8] of longint =
    (2, 0, 3, 7, 5, 8, 6, 4, 2);
begin
  j := 1;
  for i := 0 to 6 do
    j := a[j];
  write(a[j]);
end.
```

**C Version**

```
#include <stdio.h>
int i, j;
int a[9] = {2, 0, 3, 7, 5, 8, 6, 4, 2};
int main() {
  j = 1;
  for (i = 0; i <= 6; i++) {
    j = a[j];
  }
  printf("%d", a[j]);
  return 0;
}
```

- A. 2
- B. 0
- C. 8
- D. 5

21. What is the output of the following program?

**Pascal Version**

```
var
  i, j:longint;
  a:array [0..8,0..1] of longint =
    ((1, 4), (3, 7), (6, 2), (7, 3), (0, 1),
     (8, 5), (8, 7), (4, 8), (4, 7));
begin
  j := 0;
  for i := 0 to 10007 do
    j := a[j][j mod 2];
  write(j);
end.
```

**C Version**

```
#include <stdio.h>
int i, j;
int a[9][2] = {{1, 4}, {3, 7}, {6, 2}, {7, 3},
               {0, 1}, {8, 5}, {8, 7}, {4, 8}, {4, 7}};
int main() {
  j = 0;
  for (i = 0; i <= 10007; i++) {
    j = a[j][j%2];
  }
  printf("%d", j);
  return 0;
}
```

- A. 2
- B. 8
- C. 0
- D. 3

22. What is the output of the following program?

#### Pascal Version

```
function fact(n:longint):longint;
begin
  if (n = 0) then
    fact:=2
  else if ((n mod 3) = 0) then
    fact:=fact(n-1)*3
  else if ((n mod 2) = 0) then
    fact:=fact(n-1)*2
  else
    fact:=fact(n-1)*1
end;
begin
  write(fact(10))
end.
```

#### C Version

```
#include <stdio.h>
int fact(int n) {
  if (n == 0) {
    return 2;
  } else if (n%3 == 0) {
    return fact(n-1)*3;
  } else if (n%2 == 0) {
    return fact(n-1)*2;
  } else {
    return fact(n-1)*1;
  }
}
int main() {
  printf("%d", fact(10));
  return 0;
}
```

- A. 864
- B. 1728
- C. 2592
- D. None of the above

23. A compiler can compile source codes into program. But what compiles the compiler (which is a program itself)? Of course we would need a compiler for this job.

Source code A is the source code of *Compiler A*. When we compile Source code A by *Compiler A*, we will get back *Compiler A*. Source code B is the source code of *Compiler B*. When we compile Source code B by *Compiler A*, we will get *Compiler B*. Which of the followings must be true? Note that we call two files identical if they are bit-by-bit identical.

- i. *Compiler A* and *Compiler B* are compilers for the same programming language.
- ii. If we compile Source code A by *Compiler B*, the output is identical to *Compiler A*.
- iii. If we compile Source code B by *Compiler B*, the output is identical to *Compiler B*.

- A. None of the statements
- B. i only
- C. ii only
- D. iii only

24. We now modify Source code A to a new version -- Source code C. When we compile Source code C by *Compiler A*, we get *Compiler C*. *Compiler A* and *Compiler C* are compilers for the same programming language, and *Compiler C* always give faster programs compared to *Compiler A*, which means that when the same source code is compiled by the two compilers (and no compilation error occurs), and then the same input are fed to the two programs, the outputs of the two programs are the same (if no runtime error occurs), and the program compiled by *Compiler C* runs faster than that compiled by *Compiler A* (if both programs terminates) for any input. When we compile Source code A by *Compiler C*, we get *Compiler A2*. When we compile Source code C by *Compiler C*, we get *Compiler C2*. Which of the followings must be true?

- i. *Compiler A2* runs faster than *Compiler A* (it needs shorter time to compile any source code).
- ii. *Compiler A2* runs faster than *Compiler C* (it needs shorter time to compile any source code).
- iii. *Compiler C2* runs faster than *Compiler A* (it needs shorter time to compile any source code).

- A. i only
- B. iii only
- C. i and iii only
- D. i, ii and iii

25. Consider the following program segment:

**Pascal Version**

```
[I]
begin
  write('1');
  [J]
  begin
    [K]
    begin
      write('3')
    end;
    write('2')
  end
end
```

**C Version**

```
[I] {
  printf("1");
  [J] {
    [K] {
      printf("3");
    }
    printf("2");
  }
}
```

We would fill in each space ([I], [J] and [K]) by one of the following lines:

**Pascal Version**

```
L: for i := 0 to 1 do
M: for j := 0 to 2 do
N: for k := 0 to 3 do
```

**C Version**

```
L: for (i = 0; i <= 1; i++)
M: for (j = 0; j <= 2; j++)
N: for (k = 0; k <= 3; k++)
```

Which assignment produces the smallest output? (Treat the output as a decimal number.)

- |    |     |     |     |
|----|-----|-----|-----|
|    | [I] | [J] | [K] |
| A. | L   | M   | N   |
| B. | M   | N   | L   |
| C. | N   | L   | M   |
| D. | N   | M   | L   |

**Section B (20 marks)**

The blanks are labeled from A to O. Please fill in the blanks on the answer sheet.

Except otherwise specified, two marks for each correct blank. No marks will be deducted for wrong answers.

**Note:**

- (1) You must not use the `?:` operator in C.
- (2) You must not use any library function unless the appropriate library(s) is/are included.
- (3) You can write only one character in each box on the answer sheet.
- (4) No answer with length greater than the designated number of boxes will be accepted.

1. Convert the following hexadecimal number into octal number:

FACADE(16) →     A    (8) (1 mark)

Convert the following binary number into hexadecimal number:

1010010000111101(2) →     B    (16) (1 mark)

2. "Nonopus card" is a form of electronic money. By inputting the amount of its deposit into the following program, it automatically detects if the deposit is positive. If not, \$200 is (repeatedly) added until the deposit becomes positive. Then it outputs the final deposit. You may assume that the input  $x$  must be an integer in  $[-2000, 2000]$ . Complete the program.

**Pascal Version**

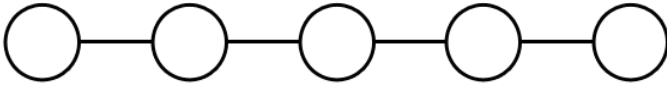
```
var
  x:longint;
begin
  read(x);
  if (x > 0) then
    write(x)
  else
    write(    C    );
end.
```

**C Version**

```
#include <stdio.h>
int x;
int main(){
  scanf("%d", &x);
  if (x > 0)
    printf("%d", x);
  else
    printf("%d",     C    );
  return 0;
}
```

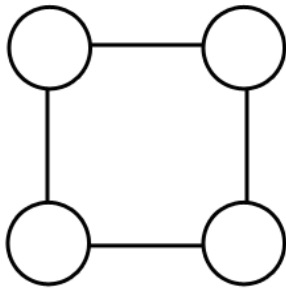
C: 2 marks

3. For a given graph, we put 1, 2, 3 or 4 into each circle such that every two adjacent circles have different numbers. We call such an arrangement "4-numbering". Two 4-coloring arrangements are different if any pair of the corresponding numbers do not match.



For the graph above, how many possible 4-numbering arrangements are there?

Answer:   D   (1 mark)



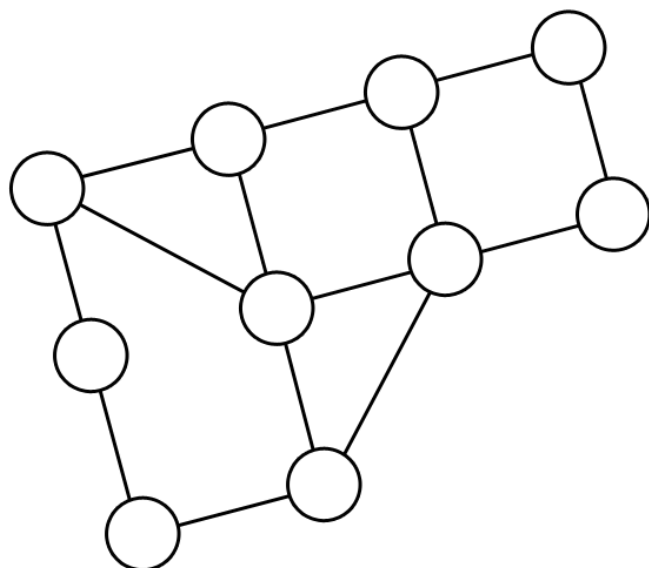
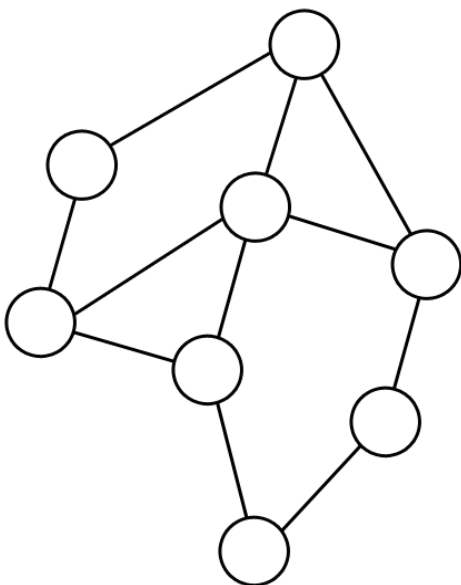
For the graph on the above, how many possible 4-numbering arrangements are there?

Answer:   E   (1.5 marks)

It is known that the graph on the left has 2,332 different 4-numbering arrangements.

Now, how many different 4-numbering arrangements are there for the graph on the right?

Answer:   F   (2 marks) **Cancelled. The original question's number was incorrect.**



4. Consider the following program:

#### Pascal Version

```

11 var
12   i, n:longint;
13 begin
14   write('Input integer n: ');
15   read(n);
16   for i := 1 to 45 do
17   begin
18     if (i*(i+1) = n*2) then
19     begin
20       write('n is a triangular number');
21     end
22   end;
23   write('n is not a triangular number');
24 end.
```

#### C Version

```

51 #include <stdio.h>
52 int n, i;
53 int main() {
54   printf("Input integer n: ");
55   scanf("%d", &n);
56   for (i = 1; i <= 45; i++) {
57     if (i*(i+1) == n*2) {
58       printf("n is a triangular number");
59     }
60   }
61   printf("n is not a triangular number");
62   return 0;
63 }
```

The above program determines whether input  $n$  is a triangular number. You may assume that  $n$  is an integer in  $[1, 1000]$ . However there is one error. Determine the type of error (compilation, run-time or logic), line number and then correct it.

Type of error:   G   (1 mark, check  the appropriate answer)

Line number:   H   (1 mark)

Correct the line:   I   (2 marks)

5. Assume function/procedure `swap(i, j)` swaps the values of `a[i]` and `a[j]`.  
 Say if `a[3] = 5` and `a[6] = 1`, after `swap(3, 6)`, `a[3]` becomes 1 and `a[6]` becomes 5.

Complete the following program segment so that it sorts array `a[9]` (**Pascal**: `a[0..8]`) in ascending order.

**Pascal Version**

```
p := 0;
while (p < 8) do
begin
  if a[p] > a[p+1] then
  begin
        J    ;
    if (    K1    ) then
          K2    ;
  end
  else
  begin
        L    ;
  end
end;
end;
```

**C Version**

```
p = 0;
while (p < 8) {
  if (a[p] > a[p+1]) {
        J    ;
    if (    K1    )
          K2    ;
  } else {
        L    ;
  }
}
```

J: 1 mark      K1 and K2: 2 marks if both correct      L: 1 mark

6. What is the output of the following program?

**Pascal Version**

```
var
s:string;
i, j:longint;
begin
s := '..>..<>..';
for i := 1 to 3 do
begin
  for j := 1 to 9 do
  begin
    if (s[j+1] = '<') then
    begin
      s[j] := '<';
      s[j+1] := '.';
    end;
    if (s[10-j] = '>') then
    begin
      s[11-j] := '>';
      s[10-j] := '.';
    end;
  end;
end;
writeln(s);
end
end.
```

**C Version**

```
#include <stdio.h>
int main() {
  int i, j;
  char s[] = "..>..<>..";
  for (i = 1; i <= 3; i++) {
    for (j = 0; j <= 8; j++) {
      if (s[j+1] == '<') {
        s[j] = '<';
        s[j+1] = '.';
      }
      if (s[8-j] == '>') {
        s[9-j] = '>';
        s[8-j] = '.';
      }
    }
    printf("%s\n", s);
  }
  return 0;
}
```

- Line 1:     M     (1 mark)  
 Line 2:     N     (1 mark)  
 Line 3:     O     (1.5 marks)