Assume that all variables without declaration shown in the following program segments have already been declared properly. Integers (`integer/int`) and long integers (`longint/long`) in problem statements are 16-bit signed variables and 32-bit signed variables respectively. Assume all the programs are compiled properly without using any compiler flag (except the "`-o`" option in C).

Truth tables

a	*b*	***a* XOR *b***	***a* XNOR *b***
F	F	F	T
F	T	T	F
T	F	T	F
T	T	F	T

Section A (22 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 statement(s) are true?

(i) A program compiled to run in Mac OS can be run in Windows directly as well.
(ii) A program cannot be compiled if it contains logic errors.

A. None of the above
B. (i) only
C. (ii) only
D. (i) and (ii)

2. Which of the following statement(s) are true?

i) It requires at least *n* bytes to store a *n*-digit decimal number.
ii) The mathematical constant π cannot be exactly represented in binary.

A. None of the above
B. (i) only
C. (ii) only
D. Both (i) and (ii)

3.

Average pounds of seafood obtained per boat:

Number of fish boats sent	Sea A	Sea B	Sea C
1	105	120	130
2	95	110	100
3	90	100	70

Base on the above information, what is the highest amount of seafood you can achieve if you have five fish boats if you must send at least one boat to each Sea?

A. 535 pounds
B. 550 pounds
C. 575 pounds
D. 585 pounds

4. p@q returns true if and only if p is true and q is false, otherwise it returns false. Simplify the following expression:

```
(p or q) @ ((not p) and q)
```

A. p
B. q
C. p and q
D. p xor q

5. Consider the following program.

Pascal Version

```
var a:array[0..5] of integer;
var i,j,temp:integer;
begin
   for i:=0 to 5 do a[i]:=i;
   for j:=1 to 2012 do
      for i:=0 to 4 do
      begin
         temp:=a[i];
         a[i]:=a[i+1];
         a[i+1]:=temp;
      end;
   for i:=0 to 5 do write(a[i]);
end.
```

C Version

```
#include <stdio.h>
int i,j,a[6],temp;
int main() {
   for (i=0; i<6; i++) a[i]=i;
   for (j=1; j<=2012; j++)
      for (i=0; i<5; i++) {
         temp=a[i];
         a[i]=a[i+1];
         a[i+1]=temp;
      }
   for (i=0; i<6; i++) printf("%d",a[i]);
}
```

What is the output of the following program?

A. 012345
B. 123450
C. 501234
D. 543210

6. Consider the following program.

Pascal Version

```
for i:=1 to 100 do
begin
   if (i mod 2 = 0) then write('*')
   else
   if (i mod 3 = 0) then write('*');
   if (i mod 4 = 0) then write('*');
end;
```

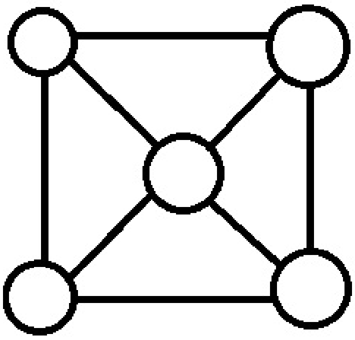
C Version

```
for (i=1; i<=100; i++)
{
   if (i%2==0) printf("*");
   else
   if (i%3==0) printf("*");
   if (i%4==0) printf("*");
}
```

How many “*” does the following program segment output?

A. 92
B. 93
C. 94
D. 95

7. Circles are neighboring if they are connected by a straight line. In the following figure, what is the minimum number of circles needed to be painted into red such that for each circle, there is at least 1 red neighboring circle?



A. 1
B. 2
C. 3
D. 4

8. There are 7 cities. What is the minimum number of unidirectional roads connecting pairs of cities needed to be constructed such that for every city, there exists a path from that city going to any other cities?

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

9. Which of the following character a valid identifier (variable name) must not contain?

A. o
B. 0
C. -
D. _

10. Concatenate 1 to 100 in binary numbers, then the resulting sequence starts with 11011100101... What is the maximum number of consecutive ‘1’s in the sequence?

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

For problems 11 and 12, consider the following clauses only.

If A happens, then at least one of B or C must happen.
If B happens, then at least one of C or D must happen
....
If X happens, then at least one of Y or Z must happen.
If Y happens, then Z must happen.
Now, A happens.

11. What is the maximum number of events to happen?

A. 1
B. 25
C. 26
D. Infinity

12. What is the minimum number of events to happen?

A. 1
B. 12
C. 13
D. 14

13. There are 20 HKOI contestants in a classroom, in which the 20 seats are organized in 4 rows and 5 columns. If a contestant sits in front of, behind, to the left of or to the right of another contestant, we call them "neighbors". Which of the following statements is true?

A. If there is exactly one Pascal user in each column, then there is a row with exactly two Pascal users.
B. If there is exactly one Pascal user in each column, then there must be two Pascal users who are neighbors
C. If there are exactly 11 Pascal users, then there must be two Pascal users who are neighbors.
D. It is possible that each of the rows and columns contains an odd number of Pascal users.

14. Consider a data structure which could be a stack or a queue. Initially it contains 10 blue balls. You can perform two operations on the data structure:

Add: Push (stack) or enqueue (queue) a red ball
Remove: Pop (stack) or dequeue (queue) a ball. If it is red, you gain 1 score. You cannot perform "remove" if the data structure is empty.

Now we are going to perform "add" 12 times and "remove" 20 times in any order (e.g. we may perform "add" 5 times, "remove" 20 times, then "add" 7 times). Which of the following statements is correct after the all operations, regardless of the order of the operations?

A. If the data structure is a stack, the final score will always be 10.
B. If the data structure is a stack, the final score will always be 12.
C. If the data structure is a queue, the final score will always be 10.
D. If the data structure is a queue, the final score will always be 12.

15. What is the output of the following program?

Pascal Version

```
var x,y : integer;

begin
     x := 1;
     y := 1;
     while (y-x<=2012) do
          begin
               x := x+x;
               y := x+y;
          end;
     write(x+y,' ',x-y);
end.
```

C Version

```
#include <stdio.h>
int x,y;
int main(){
   x=1;
   y=1;
   while (y-x <= 2012){
      x=x+x;
      y=x+y;
   }
   printf("%d %d",x+y,x-y);
}
```

A. `1535 -511`
B. `3071 -1023`
C. `6143 -2047`
D. `12287 -4095`

16. What is the output of the following program?

Pascal Version

```
var
   i,j,k,x : integer;
begin
   x := 0;
   for i := 1 to 5 do
       for j := i to 5 do
           for k := j to 5 do
               x := x+1;
   write(x);
end.
```

C Version

```
#include <stdio.h>
int i,j,k,x;
int main(){
  x=0;
  for(i=1; i<=5; i++)
      for(j=i; j<=5; j++)
          for(k=j; k<=5; k++)
              x = x+1;
  printf("%d",x);
}
```

A. 34
B. 35
C. 36
D. 37

17. What is the purpose of the following program?
(Assume the inputting numbers are all positive integers less than 100.)

Pascal Version

```
var
  n,x,i : integer;
  a : array[0..99] of integer;
begin
  for i := 1 to 99 do a[i] := 0;
  read(n);
  for i := 1 to n do
  begin
     read(x);
     a[x] := a[x]+1;
  end;
  for i := 1 to 99 do
     if (a[x] > 0) writeln(i);
end.
```

C Version

```
#include <stdio.h>
int a[100],n,x,i;
int main(){
  for(i=1;i<=99;++i) a[i]=0;
  scanf("%d",&n);
  for(i=1;i<=n;i++){
     scanf("%d",&x);
     a[x]=a[x]+1;
  }
  for(i=1;i<=99;++i)
     if (a[i]>0) printf("%d\n",i);
}
```

A. Output how many times each of the input numbers occurs
B. Sort all the input numbers in non-decreasing order
C. Output the numbers 1 to 99
D. Eliminate all repeated numbers in the input, and then sort the numbers in non-decreasing order

18. Consider the following program.

Pascal Version

```
var  i:integer;
     a:array[0..9] of ^integer;
     b:array[0..9] of integer;

begin
   for i:=0 to 9 do b[i] := i;
   for i:=0 to 9 do a[i] := @b[i];
   for i:=0 to 4 do swap(b[2*i], b[2*i+1]);
   for i:=0 to 4 do swap(a[i], a[9-i]);
   for i:=0 to 4 do swap(a[2*i]^, a[2*i+1]^);

   for i:=0 to 9 do write(a[i]^);
   writeln;
end.
```

C Version

```
int i;
int *a[10], b[10];
int main(){
   for (i=0; i<10; i++) b[i] = i;
   for (i=0; i<10; i++) a[i] = &b[i];
   for (i=0; i<5; i+=2)
     swap(b[2*i], b[2*i+1]);
   for (i=0; i<5; i++) swap(a[i], a[9-i]);
   for (i=0; i<5; i+=2)
     swap(*a[2*i], *a[2*i+1]);
   for (i=0; i<10; i++) printf("%d ", *a[i]);
   printf("\n");
   return 0;
}
```

Assume `swap(x, y)` is a function that swaps the contents of x and y. What is the output of the program?

A. 0 1 2 3 4 5 6 7 8 9
B. 9 8 7 6 5 4 3 2 1 0
C. 1 0 3 2 5 4 7 6 9 8
D. 8 9 6 7 4 5 2 3 0 1

19. How many '*' will the following program output?

Pascal Version

```
var x:integer;
procedure put();
begin
     write('*');
     if (x <= 5) then
     begin
          x := x + 1;
          put();
          x := x + 1;
          put();
     end;
end;
begin
     x := 0;
     put();
end.
```

C Version

```
#include <stdio.h>
int x;
void put(){
   printf("*");
   if (x <= 5){
      x++;
      put();
      x++;
      put();
   }
}
int main(){
   x = 0;
   put();
   return 0;
}
```

A. 12
B. 13
C. 14
D. 15

20. At most how many Black Friday can there be in 2 consecutive common years? (Common year means a year with 365 days. Black Friday is a Friday which lies on the 13th day of the month)

A. 4
B. 5
C. 6
D. 7

21. Which of the following expressions correctly determine if Y is a leap year?

Pascal Version

```
1) (Y mod 4=0)and((Y mod 100<>0)or(Y mod 400=0))
2) ((Y mod 4=0)and(Y mod 25<>0))or(Y mod 400=0)
3) (Y mod 4=0)and((Y mod 25<>0)or(Y mod 100=0))
```

C Version

```
1) Y%4==0 && (Y%100!=0 || Y%400==0)
2) (Y%4==0 && Y%25!=0) || Y%400==0
3) Y%4==0 && (Y%25!=0 || Y%100==0)
```

A. 1 only
B. 2 only
C. 1 and 2
D. All of them

22 Consider the following program.

Pascal Version

```
var b : array[0..1000] of integer;
    i,j,k : integer;
begin
    for i := 0 to 1000 do
        b[i] := 0;
    for i := 2 to 1000 do
        begin
        if b[i] = 0 then
            begin
            writeln(i);
            j := ___@___;
            while (j<=1000) do
                begin
                b[j] := true;
                j := j+i;
                end;
            end;
        end;
end.
```

C Version

```
int b[1001];
int i,j,k;

int main(){
    for (i=0;i<=1000;++i)
        b[i] = 0;
    for (i=2;i<=1000;++i) {
        if (b[i]==0){
            printf("%d\n",i);
            j = ___@___;
            while (j<=1000){
                b[j] = 1;
                j+=i;
            }
        }
    }
}
```

To print all prime under 1000, which replacement for ___@___ give the best performance?

A. `i`
B. `i+i`
C. `i*i`
D. `i*(i+1)`

END OF SECTION A

Section B (23 marks)

The blanks are labeled from A to J. Please fill in the blanks on the answer sheet.
Except otherwise specified, two marks for each correct answer. 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 on each box on the answer sheet.
(4) No answer with length greater than the designated number of boxes will be accepted.

1. Let *m*(a,b,c) be a function that gives the median of the three integers a, b and c.
   For example, *m*(2,1,3) = 2. Answer A and B using this function.
   The characters "?", ":", "<" and ">" are not allowed.

   A. Write an expression which gives the maximum of two non-negative integers x and y.
   B. Write an expression which gives the maximum of two integers x and y

2. For problem C to E, please consider the following situation.

   Alice has written a program to find all the prime numbers in the range [1,25]. Complete the following program segment below.

Pascal Version

```
var is_prime : array[0..5] of integer;
    prime : integer;
......
is_prime[2]:=1;
is_prime[3]:=1;
is_prime[4]:=0;
is_prime[5]:=_______(C)________;
for i:=____(D)____ to 25 do
begin
  prime:=1;
  for j:=2 to 5 do
    if (________(E)________) then
       prime:=0;
    if prime=1 then writeln(i);
end;
```

C Version

```
int is_prime[6];
int prime;
......
is_prime[2]=1;
is_prime[3]=1;
is_prime[4]=0;
is_prime[5]= _______(C)________;
for (i=__ _(D)_____; i<=25; i++) {
  prime=1;
  for (j=2; j<=5; j++)
    if (_________(E)________)
      prime=0;
  if (prime==1) printf("%d\n",i);
}
```

3. If we add all of its digits for a certain positive integer, we could get a new positive integer. This procedure could be repeated, until the final result has only one digit.

For example, 1234 would produce a new integer 10 (1+2+3+4), and finally 1 (1+0). Janet wants to write a program to input a 5-digit positive integer less than 32768, and output the final single digit after performing the above-mentioned procedure.

Here is her code:

Pascal Version

```
var x : integer;
begin
     read(x);
     write(x-((x+1) div 9)*9);
end.
```

C Version

```
#include <stdio.h>
int x;
int main(){
   scanf("%d",&x);
   printf("%d", x-(x+1)/9*9);
}
```

However, she realizes that her program does not work for some cases.
Help Janet write down in F a counter-example which is within the input range.

Scoring:
2 marks if the written integer is a correct counter-example
3 marks if the written counter-example is the maximum possible integer

4. There is a robot in a grid. You can use a command to control the robot:

∧ : up
∨ : down
< : left
> : right
[*Command*]*num* : repeat *Command* for *num* times, where *num* is an integer from 1 to 9 (inclusive).

Formally, '∧', '∨', '<', '>' are **moves**. One or more **moves** forms a **Command**. (*Command*)*num* is also a **command**. Concatenation of two **commands** is also a **command**.

Example of a command: ∧[>]<∨]3]2

If a **move** will result the robot out of boundary or hit an obstacle, the robot will ignore that move.
In the following problems, you need to write a command to fulfill the task, where the robot should make **less than 1000 moves** and the length of the command must **not exceed than the provided space**.

```
R........
.........
.........
.........
.........
.........
.........
.........
.........
```

R is the initial position of the robot.

G. Write a command to make the robot visit each grid at least once. (1 mark)

H. Assume that the robot can be start at any position in the grid, Write a command to make the robot visit each grid at least once. (1.5 marks)

```
R........
.........
.........
.........
....*....
.........
.........
.........
.........
```

* is an obstacle

I. Write a command to make the robot visit each '**.**' grid at least once. (2 marks)

J. Assume that the robot can be start at any position in the grid, write a command to make the robot visit each '**.**' grid at least once. (2.5 marks)

K. Assume that the robot can be start at any position in the grid and the obstacle can also appear anywhere in the grid, Write a command to make the robot visit each '**.**' grid at least once. (3 marks)

END OF PAPER