

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).

**Notations**

For any real number  $x$ ,

- (1)  $\lfloor x \rfloor$  denotes the largest integer not greater than  $x$ .
- (2)  $\lceil x \rceil$  denotes the smallest integer not less than  $x$ .
- (3)  $|x|$  denotes the absolute value of  $x$ .

For non-negative integer  $n$ , its factorial  $n! = 1 \times 2 \times 3 \times \dots \times (n-1) \times n$ ; and  $0! = 1$ .

**Truth tables**

$a$	$b$	$a \text{ XOR } b$	$a \text{ XNOR } b$
F	F	F	T
F	T	T	F
T	F	T	F
T	T	F	T

**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. At least how many times of hand shaking are required among 100 people so that everyone can shake hands at least once?
  - A. 5050
  - B. 4950
  - C. 100
  - D. 50

2. Euclidean distance between 2 points  $(x_1, y_1)$  and  $(x_2, y_2)$  is defined as  $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ .  
 Manhattan distance between 2 points  $(x_1, y_1)$  and  $(x_2, y_2)$  is equal to  $|x_1 - x_2| + |y_1 - y_2|$

Which of the following statement(s) about these two distances must be **TRUE**?

- (i) For a triangle ABC, the sum of Euclidean distances between A and B and that between B and C is greater than that between A and C.
- (ii) For a triangle ABC, the sum of Manhattan distances between A and B and that between B and C is greater than that between A and C.

(Assume that A, B and C do not form a straight line.)

- A. (i) only
  - B. (ii) only
  - C. (i) and (ii)
  - D. None of the above
3. A positive integer  $x$  has 3 digits. Which of the following statement(s) must be **TRUE**?
    - (i)  $x^2$  is a 5-digit integer.
    - (ii) If  $x=y^2$ , then  $y$  is a 2-digit integer.
    - A. (i) only
    - B. (ii) only
    - C. (i) and (ii)
    - D. None of the above

4. What is the output of the following program?

**Pascal Version**

```
var
i,j:integer;
  a:array[1..16] of boolean;
begin
  for i:=1 to 16 do a[i]:=false;
  for i:=1 to 16 do
  begin
    j:=i;
    while j<=16 do
    begin
      a[j]:=not a[j];
      j:=j+i;
    end;
  end;
  writeln(a[8], ' ',a[16]);
end.
```

- A. TRUE TRUE
- B. TRUE FALSE
- C. FALSE TRUE
- D. FALSE FALSE

**C Version**

```
#include <stdio.h>
int i,j,a[17];
int main(){
  for (i=1;i<=16;i++) a[i]=0;
  for (i=1;i<=16;i++){
    j=i;
    while (j<=16){
      a[j]=1-a[j];
      j=j+i;
    }
  }
  printf("%d %d\n",a[8],a[16]);
}
```

- A. 1 1
- B. 1 0
- C. 0 1
- D. 0 0

5. What is the output of the following program?

**Pascal Version**

```
a[0]:=1; b[0]:=0;
a[1]:=0; b[1]:=4;
a[2]:=4; b[2]:=2;
a[3]:=2; b[3]:=3;
a[4]:=3; b[4]:=1;
ans := 0;
for i := 0 to 4 do
  for j := 0 to 4 do
    ans := ans+a[b[i]]+b[a[j]];
writeln(ans);
```

- A. 75
- B. 100
- C. 125
- D. 150

**C Version**

```
a[0]=1; b[0]=0;
a[1]=0; b[1]=4;
a[2]=4; b[2]=2;
a[3]=2; b[3]=3;
a[4]=3; b[4]=1;
ans=0;
for (i=0; i<5; i++)
  for (j=0; j<5; j++)
    ans+=a[b[i]]+b[a[j]];
printf("%d\n",ans);
```

6. Here is a conversation among Ada, Bob and Cindy:

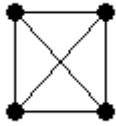
Ada: Cindy is a liar!  
 Bob: Yes! Ada is telling the truth!  
 Cindy: Don't believe in Bob! He is lying!

It is known that at least one of them is honest.  
 Who is/are honest/dishonest?

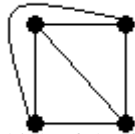
- |    |                      |           |           |
|----|----------------------|-----------|-----------|
|    | Ada                  | Bob       | Cindy     |
| A. | Honest               | Dishonest | Dishonest |
| B. | Dishonest            | Honest    | Dishonest |
| C. | Dishonest            | Dishonest | Honest    |
| D. | Cannot be determined |           |           |

7. How many pairs of prime numbers between 1 to 20 (inclusively) have their sum divisible by 4?  
 A. 10  
 B. 11  
 C. 12  
 D. 13

8. A planar graph is a graph which exists a drawing such that the edges do not intersect.  
 For example, the following graph is a planar graph.

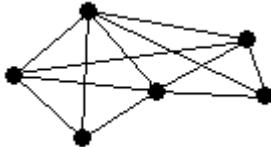


This is because it can be transformed to the graph below, without changing the connectivity of each pair of vertices:

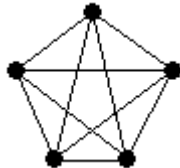


Which of the following(s) is/are planar graph(s)?

(i)



(ii)



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

9. Consider the following program segment:

**Pascal Version**

```
count:=0;
readln(a);
while (a>0) do
begin
    a:=a div 2;
    count := count+1;
end;
writeln(count);
```

**C Version**

```
count = 0;
scanf("%d",&a);
while (a>0) {
    a = a / 2;
    count = count + 1;
}
printf("%d\n",count);
```

If the output is 8 and the input is a positive integer, how many different possible inputs are there?

- A. 8  
 B. 16  
 C. 128  
 D. 256

10. Consider the following program segment:

**Pascal Version**

```
for i := 0 to n-1 do
  read(a[i]);
for i := 0 to n-1 do
  a[i]:=a[n-i-1]
for i := 0 to n-1 do
  writeln(a[i]);
```

**C Version**

```
for (i=0; i<n; i++)
  scanf("%d",&a[i]);
for (i=0; i<n; i++)
  a[i]=a[n-i-1];
for (i=0; i<n; i++)
  printf("%d",a[i]);
```

If the input is "2 0 1 0 1 1 0 6" and  $n=8$ , what is the output of the program?

- A. 20101106
- B. 60111106
- C. 20106011
- D. 60110102

11. Alice had written the program segment below for sorting an array of 5000 integers in ascending order.

**Pascal Version**

```
for i:=0 to 4999 do
  for j :=0 to 4998 do
    if a[j]>a[j+1] then
      swap(a[j],a[j+1]);
```

**C Version**

```
for (i=0; i<5000; i++)
  for (j=0; j<4999; j++)
    if (a[j]>a[j+1])
      swap(a[j],a[j+1]);
```

Peddy argued that his program was not efficient enough. He changed the initial value of  $j$  in the inner for-loop.

**Pascal Version**

```
for i:=0 to 4999 do
  for j := ___?___ to 4998 do
    if a[j]>a[j+1] then
      swap(a[j],a[j+1]);
```

**C Version**

```
for (i=0; i<5000; i++)
  for (j= ___?___; j<4999; j++)
    if (a[j]>a[j+1])
      swap(a[j],a[j+1]);
```

Which of the following should be filled in the blank such that the program is correct and consumes least time? You can assume the smallest element is not the last element.

- A. 1
- B. i
- C. i+1
- D. i\*2

12. Let  $A[0..50]$  be an array of integers. We get  $B[]$  by reversing the order of elements in  $A[]$ .

Which of the following is **NOT TRUE**?

- A.  $A[25]$  is equal to  $B[25]$ .
- B. If  $A[10]<A[20]$ , then  $B[30]>B[40]$ .
- C. The sum of  $A[15]$  to  $A[30]$  is equal to the sum of  $B[20]$  to  $B[35]$ .
- D. If  $A[40]$  is the maximum of  $A$ , then  $B[40]$  is the minimum of  $B$ .

13. Four books are going to be put into a box one by one and cover the previous books in the box. Only the book on the top of the box can be taken out. If these books are going to take out afterwards in this order: Physics, Chemistry, Maths and lastly Computer, what order should you put those books into the box?

- A. Physics  $\rightarrow$  Chemistry  $\rightarrow$  Maths  $\rightarrow$  Computer
- B. Computer  $\rightarrow$  Maths  $\rightarrow$  Chemistry  $\rightarrow$  Physics
- C. Chemistry  $\rightarrow$  Computer  $\rightarrow$  Maths  $\rightarrow$  Physics
- D. Maths  $\rightarrow$  Physics  $\rightarrow$  Chemistry  $\rightarrow$  Computer

14. You have a strip of paper with some digits written on it. You want to cut this paper into several pieces, each with some digits on it and form an integer, such that the sum of the integers in all pieces is divisible by 10. How many ways can you do that? NO leading zero is allowed. You can choose not to cut at all.

540560

For example,  $540 + 560 = 1100$  is divisible by 10

- A. 2
- B. 3
- C. 4
- D. 5

15. Consider the following program segment:

**Pascal Version**

```
readln(a);
while a>0 do
begin
  a:=a div 2;
  writeln(a mod 2);
end;
```

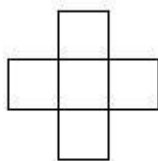
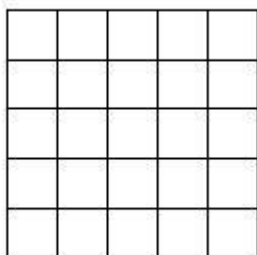
**C Version**

```
scanf("%d",&a);
while (a > 0) {
  a = a/2;
  printf("%d",a%2);
}
```

What is the output of the program if the input is 19?

- A. 11000
- B. 00011
- C. 100101
- D. 10010

16. How many '+' pieces do you need to cover every square in a 5x5 board? (The '+' pieces can overlap and can be out of boundary)



'+' piece

- A. 6
- B. 7
- C. 8
- D. 9

17. Alice claims that for all positive integer  $x$  not less than  $y$ , she can pay  $\$x$  exactly using only  $\$3$  coins and  $\$7$  coins. What is the minimum value of  $y$ ?

- A. 12
- B. 15
- C. 21
- D. 24

18. Bob will do the revision if and only if he is alone and it is night time. Also, if it is night time, he will eat chocolate. Which of the following statements must be **TRUE** if he is **NOT** eating chocolate?

- (i) He is not doing the revision.
- (ii) He is not alone.
- (iii) It is not night time now.

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

19. Suppose there are  $n$  islands in the Bob's Kingdom. Each bridge connects two islands. What is the minimum number of bridges required such that for all pairs of islands, it is possible to travel from one to another using at most 2 bridges? You can assume all bridges are bidirectional.

- A.  $n-1$
- B.  $n$
- C.  $n+1$
- D.  $2n$

20. Suppose `Rand(2)` returns 0 or 1 with equal probability.

**Pascal Version**

```
x := Rand(2);
y := Rand(2);
writeln(x-y);
```

**C Version**

```
x = Rand(2);
y = Rand(2);
printf("%d\n", x-y);
```

Which of the following(s) must be **TRUE**?

- (i) The output is 0.
- (ii) There are 3 possible outputs.
- (iii) All possible outputs have equal probability of appearing.

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

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

**Pascal Version**

```
for i:=1 to 5 do
begin
  for j := 1 to 5 do
  begin
    if abs(i-3)+abs(j-3)<=2 then
      write('*')
    else write('.');
  end;
  writeln;
end;
```

**C Version**

```
for (i=1;i<=5;i++){
  for (j=1;j<=5;j++){
    if (abs(i-3)+abs(j-3)<=2)
      printf("*");
    else pritrnf(".");
  }
  printf("\n");
}
```

A.

```
.....
.***.
.***.
.***.
.....
```

B.

```
..*..
.***.
*****
.***.
..*..
```

C.

```

.....
.***.
.*.*.
.***.
.....

```

D.

```

...*.
.*.*.
*...*
.*.*.
...*.

```

22. Given a sequence containing 10 distinct integers  $a_1, a_2, \dots, a_{10}$ , which satisfies  $a_1 > a_2, a_2 < a_3, a_3 > a_4, \dots, a_8 < a_9, a_9 > a_{10}$ . Which of the following statements must be **TRUE**?
- (i) If  $a_1 > a_3$ , then  $a_2 > a_4$ .
  - (ii) The minimum number of comparisons to find out the largest element is 4.
  - (iii)  $a_{10}$  is the smallest element.
- A. (i) only
  - B. (ii) only
  - C. (i) and (iii) only
  - D. (ii) and (iii) only
23. Which of the following statements is **NOT** always **TRUE**?
- A. If an integer  $x$  can be stored correctly using a 32-bit signed integer, then  $x^2$  can be stored correctly using a 64-bit signed integer.
  - B. If a positive integer  $x$  can be stored correctly using a 64-bit signed integer, then  $\sqrt{x}$  can be stored correctly using a 32-bit signed integer.
  - C. If a positive integer  $x$  can be stored correctly using a 32-bit signed integer, then  $2x$  can be stored correctly using a 32-bit unsigned integer.
  - D. If a positive integer  $x$  can be stored correctly using a 32-bit signed integer, then  $2x^2$  can be stored correctly using a 64-bit unsigned integer.
24. You are given a sequence of 5 numbers: 4 5 3 2 1. You can exchange any two adjacent numbers each time. At least how many exchanges do you need so that the sequence is sorted in increasing order?
- A. 7
  - B. 8
  - C. 9
  - D. 10
25. How many non-empty consecutive subsequences of the following sequence  $\{3, 10, 8, 9, 4, 6, 7, 5\}$  have an odd sum? For example, subsequence  $\{8, 9\}$  has an odd sum but subsequence  $\{4, 6, 7, 5\}$  does not.
- A. 20
  - B. 21
  - C. 22
  - D. 23

END OF SECTION A

**Section B (20 marks)**

The blanks are labeled from A to J. Please fill in the blanks on the answer sheet.  
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. You are going to modify a string containing only capital letters using the HKOI text editor. Originally, the string is on the editor and a cursor is highlighting the first character of the string.  
There are 4 kinds of command provided by the editor.

command	meaning
m?	Modify the character highlighted by the cursor to "?"
l	Shift the cursor to the position on the left of it. If the cursor is at the leftmost position, it will stay at the leftmost position
r	Shift the cursor to the position on the right of it. If the cursor is at the rightmost position, it will stay at the rightmost position
(command) <i>N</i>	Repeat the command by <i>N</i> times. For example "(rr)3" means shifting the cursor to the right by 6 times.

In the example and question below, ' \_ ' indicates the position of the cursor.

If the screen is initially shown as:

ABCDE

After executing the sequence of command "mF (r) 7mZ", the screen will be shown as:

FBCDZ

Write the sequence of command for modifying the following strings respectively.

Initial string:

TTCTCTCTCTC

Target string:

AGAGAGAGAGAG

Command:   A  

Initial string:

PQRRRRRRPQRRRRR

Target string:

ABCCCCCABCCCCA

Command:   B



2. Alice has implemented a binary search which was going to print the index of the target in array  $a[] = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$  ( $a[0] \dots a[11]$ ). However, Alice found that the program did not print the correct index.

Rewrite line with comment "#" in order to correct the program. Assume the initial value of target is between 1 and 10 inclusively.

**Pascal Version**

```

left:=1;
right:=10;
while left<right do
begin
  mid:=(left+right) div 2; // #
  if a[mid] > target then
    right := mid-1
  else
    left := mid;
end;
writeln(left);

```

**C Version**

```

left = 1;
right = 10;
while (left < right){
  mid = ( left + right ) / 2; /* # */
  if (a[mid] > target)
    right = mid - 1;
  else
    left = mid;
}
printf("%d\n", left);

```

#:   **C**  

3. A regular ruler may have integer marks from 0 to the length of the ruler. However, sometimes a ruler has some missing marks, but you can still use it to measure any integer length from 1 up to the ruler's full length. This kind of ruler is called 'sparse ruler'. An example of a sparse ruler of full length 6 is given below:



You can measure any integer length from 1 to 6. For example, you can get length 2 by measuring between 4 and 6, length 5 by measuring between 1 and 6 etc.

Create a sparse ruler of length 9 with at most 5 integer marks.

  **D**  

Create a sparse ruler of length 13 with at most 6 integer marks

  **E**

4. Alice has written the following program segment below to count the number of prime factors for a given integer  $n$ , where  $n$  is an integer between 2 and 1001 inclusively. For example,  $12=2^2 \times 3$ , hence 12 has 2 prime factors. However, Bob claims that her program is incorrect. Can you provide a valid input such that his program gives an incorrect output?

Input:  $n$

Output: Number of prime factors of  $n$

#### Pascal Version

```
ans:=1;
readln(n);
i:=2;
while i*i<=n do
begin
  if n mod i=0 then
  begin
    ans:=ans+1;
    while (n mod i=0) do
      n:=n div i;
    end;
    i:=i+1;
  end;
end;
writeln(ans);
```

#### C Version

```
ans=1;
scanf("%d",&n);
for (i=2;i*i<=n;i++){
  if (n%i==0){
    ans++;
    while (n%i==0)
      n/=i;
  }
}
printf("%d\n",ans);
```

$n =$      **F**    

5. Alice has written a program to find the median of an integer array  $A[]$  with  $n$  elements, where  $n$  is an odd number. Complete the following program segment below. Store the median in the variable `med`.

#### Pascal Version

```
for i:=1 to n do
begin
  x:=0; y:=0; z:=0;
  for j:=1 to n do
  begin
    if A[j]>A[i] then x:=x+1;
    if A[j]=A[i] then y:=y+1;
    if A[j]<A[i] then z:=z+1;
  end;
  if (    G    ) then med:=A[i];
end;
```

#### C Version

```
for (i=1;i<=n;i++){
  x=y=z=0;
  for (j=1;j<=n;j++){
    if (A[j]>A[i]) x++;
    if (A[j]==A[i]) y++;
    if (A[j]<A[i]) z++;
  }
  if (    G    ) med=A[i];
}
```

6. The following program segment is to find out how many times "hko" appears in the string variable `str` with length  $n$ . For example, the program should output "2" when `str` is "hkoihethkoi". Fill in the blanks.

**Pascal Version**

```

cnt:=0;
found:=0;
for i:=1 to n-1 do
begin
  cnt:=cnt+1;
  if (str[i]='h') and (str[i+1]='k')
  then cnt:=0;
  if (str[i]='o') and (str[i+1]='i')
  then
    if (     H     )
    then found:=found+1;
end;
writeln(found);

```

**C Version**

```

cnt = 0;
found = 0;
for(i=0; i<n-1; i++){
  cnt++;
  if(str[i]=='h' && str[i+1]=='k')
    cnt = 0;
  if(str[i]=='o' && str[i+1]=='i')
    if(     H     )
      found++;
}
printf("%d\n", found);

```

Hint: Consider the case when `str` is "ioi".

7. Consider the following function:

**Pascal Version**

```

function f(x:integer):integer;
var
  i, sum, a, b:integer;
begin
  sum:=0;
  a:=     I    ;
  b:=     I    ;
  for i:=1 to x do
    sum:=sum+(a*i) div b;
  f:=sum;
end;

```

**C Version**

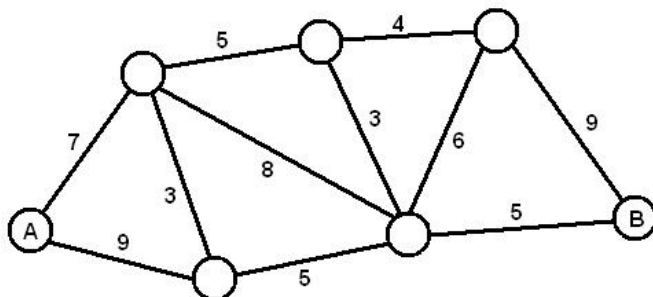
```

int f(int x){
  int i, sum = 0;
  int a =     I    ;
  int b =     I    ;
  for (i=1; i<=x; ++i)
    sum+=(a*i)/b;
  return sum;
}

```

Give a pair of values of `a` and `b` such that the return value is 17 while the parameter `x` is 5.

8. Alice is going to destroy Bob's Kingdom. More precisely, she wants to destroy some bridges such that island A and island B are disconnected. Assume the numbers written on the edges in the diagram below are the costs of the corresponding bridges. What is the minimum cost to achieve this?



Minimum cost =     J    

**END OF PAPER**