



C++ 二级

2026 年 09 月

1 单选题（每题 2 分，共 30 分）

题号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
答案	C	B	A	D	A	C	B	D	B	D	D	C	A	A	A

第 1 题 第二届世界人形机器人运动会开幕式上，机器人方阵通过控制程序来完成整场表演。表演前，工程师们使用集成开发环境（IDE）编写并调试方阵的控制程序。下列关于 IDE 功能的描述，错误的是（ ）。

- ☐ A. 可以提供代码自动补全，提高编程效率
- ☐ B. 可以提供辅助信息，帮助排查程序逻辑错误
- ☐ C. 可以直接控制机器人电机转动
- ☐ D. 支持语法高亮，使代码更易阅读

第 2 题 计算机 CPU 中的高速缓冲存储器（Cache）主要为提升系统性能而设计。下列关于 Cache 的说法中正确的是（ ）。

- ☐ A. Cache 的容量比主存大，速度比主存快
- ☐ B. Cache 通常用于暂存 CPU 频繁访问的数据
- ☐ C. Cache 中的数据断电后不会丢失，属于非易失性存储器
- ☐ D. Cache 的访问速度比 CPU 寄存器还要快

第 3 题 执行如下 C++ 代码，其输出是（ ）。

```
1 | cout << (3 == 3.0);
```

- ☐ A. 1
- ☐ B. 1.0
- ☐ C. 0
- ☐ D. -1

第 4 题 执行如下 C++ 代码，其输出是（ ）。

```
1 | cout << (bool(5) + bool(-5) + bool(5.0));
```

- ☐ A. 0
- ☐ B. 1
- ☐ C. 2
- ☐ D. 3

第5题 判断年份 `N` 是否为闰年的规则是：能被 `4` 整除但不能被 `100` 整除，或者能被 `400` 整除。以下 C++ 表达式中，能正确判断 `N` 是否为闰年的是（ ）。

- ☐ A. `(N % 4 == 0 && N % 100 != 0) || (N % 400 == 0)`
- ☐ B. `N % 4 == 0 && N % 100 == 0 || N % 400 == 0`
- ☐ C. `(N / 4 == 0 && N / 100 != 0) || (N / 400 == 0)`
- ☐ D. `(N % 4 == 0 && N % 100 != 0) && (N % 400 == 0)`

第6题 执行如下 C++ 代码，其输出是（ ）。

```
1 int tot, i;
2 tot = 0;
3 for (i = 0; i < 10; i++)
4     if (i)
5         tot += 1;
6 cout << tot;
```

- ☐ A. 0
- ☐ B. 8
- ☐ C. 9
- ☐ D. 10

第7题 执行如下 C++ 代码，当输入 `4` 时，其输出是（ ）。

```
1 int n, tot, i, j;
2 cin >> n;
3 tot = 0;
4 for (i = 1; i < n; i++)
5     for (j = 1; j < i; j++)
6         tot += 1;
7 cout << tot;
```

- ☐ A. 1
- ☐ B. 3
- ☐ C. 6
- ☐ D. 10

第8题 下面的 C++ 代码执行后其输出是（ ）。

```
1 int tnt, i;
2 tnt = 0;
3 for (i = 0; i < 10; i++)
4     if (i % 3)
5         tnt += i;
6 cout << tnt;
```

- ☐ A. 12
- ☐ B. 18
- ☐ C. 22
- ☐ D. 27

第9题 下面的 C++ 代码之后的输出是（ ）。

```

1  int tot = 0, i;
2  for (i = -5; i < 5; i++)
3      if (not(i % 3 != 0))
4          tot += 1;
5  cout << tot;

```

- ☐ A. 2
- ☐ B. 3
- ☐ C. 7
- ☐ D. 9

第 10 题 有关如下 C++ 代码描述，变量都是整型，错误的是（ ）。

```

1  int n, limit, i;
2  cout << "请输入一个正整数: ";
3  cin >> n;
4  if (n <= 1)
5      printf("%d非素数", n);
6  else if (n == 2)
7      printf("2是素数");
8  else {
9      limit = sqrt(n) + 1;
10     for (i = 2; i < limit; i++)
11         if (n % i == 0) {
12             printf("%d非素数", n);
13             break;
14         }
15     if (i >= limit)
16         printf("%d是素数", n);
17 }

```

- ☐ A. `n <= 1` 条件用于判断输入 0、1 以及负数时的情况，不能省略。
- ☐ B. `n == 2` 条件及其相关代码省略，程序的输出结果没有区别。
- ☐ C. `break` 语句不可以省略，否则非素数还将执行最后一行的语句。
- ☐ D. `limit = sqrt(n) + 1` 语句中的 1 可以修改为其他大于等于 1 的正整数，执行结果不会有错误，但循环次数增多，效率有所影响。

第 11 题 执行以下 C++ 代码后，最终的输出结果是（ ）。

```

1  int total, i, j;
2  total = 0;
3  i = 1;
4  while (i <= 3) {
5      for (j = 1; j < i + 1; j++)
6          total += i * j;
7      i += 1;
8  }
9  cout << total;

```

- ☐ A. 9
- ☐ B. 11
- ☐ C. 23
- ☐ D. 25

第 12 题 下面 C++ 代码执行后其输出是（ ）。

```

1  int count, i, j;
2  count = 0;
3  for (i = 1; i < 4; i++)
4      for (j = 0; j < i; j++) {
5          if (j % 3 == 0)
6              continue;
7          count += 1;
8          if (i * j >= 2)
9              break;
10     }
11     printf("%d %d %d", i, j, count);

```

- ☐ A. 2 3 2
- ☐ B. 3 0 2
- ☐ C. 4 1 2
- ☐ D. 4 2 3

第 13 题 如下的 C++ 代码执行时输入 12 将输出如下数字图形。注意：最后一行之前没有空格。横线处应填入的代码是（ ）。

```

1      0
2      12
3      345
4      6789
5      01234
6      567890
7      1234567
8      89012345
9      678901234
10     5678901234
11     56789012345
12     678901234567

```

```

1  int line, i, j, k;
2
3  i = 0;
4  cout << "请输入行数: ";
5  cin >> line;
6  for (j = 0; j < line; j++) {
7      for (k = 0; k < _____; k++) // L1
8          cout << " ";
9      for (k = 0; k < j + 1; k++) {
10         cout << i;
11         i += 1;
12         if (i >= 10)
13             i = 0;
14     }
15     cout << endl;
16 }

```

- ☐ A. line - j - 1
- ☐ B. line - j
- ☐ C. line
- ☐ D. line + 1

第 14 题 小北的身高是 1.65 米。班上有 n 个小朋友，小北想和身高最接近自己的人交朋友。如果有多个小朋友与小北的身高差相同，小北想和其中身高较矮的第一个小朋友做朋友。横线处应该填入的是（ ）。

```

1  int n, i;
2  float target, h, best_height, min_diff, diff;
3
4  cout << "请输入小朋友的人数: ";
5  cin >> n;
6  target = 1.65;
7  best_height = 0.0;
8  min_diff = 100.0;
9
10 for (i = 0; i < n; i++) {
11     cout << "请输入第 " << i << " 个小朋友的身高: ";
12     cin >> h;
13     diff = abs(h - target);
14
15     if (_____) { // L1
16         min_diff = diff;
17         best_height = h;
18     }
19 }
20 cout << "小北的朋友身高为: " << best_height;

```

- ☐ A. `diff < min_diff || (diff == min_diff && h < best_height)`
- ☐ B. `diff <= min_diff && h < best_height`
- ☐ C. `diff < min_diff && h < best_height`
- ☐ D. `diff < min_diff || h < best_height`

第15题 小北的身高是 1.65 米。班上有 n 个小朋友，小北想和身高最接近自己的人交朋友。如果有多个小朋友与小北的身高差相同，小北想和其中身高较矮的第一个小朋友做朋友。横线处应该填入的是（ ）。

```

1  int n, i;
2  float target, h, best_height, min_diff, diff;
3
4  cout << "请输入小朋友的人数: ";
5  cin >> n;
6  target = 1.65;
7  best_height = 0.0;
8  min_diff = 100.0;
9
10 for (i = 0; i < n; i++) {
11     cout << "请输入第 " << (i + 1) << " 个小朋友的身高: ";
12     cin >> h;
13
14     if (h > target)
15         diff = h - target;
16     else
17         diff = target - h;
18
19     if (diff < min_diff) {
20         min_diff = diff;
21         best_height = h;
22     } else if (diff == min_diff) {
23         _____
24         _____
25     }
26 }
27
28 cout << "小北的朋友身高为:" << best_height;

```

☐ A.

```
1 | if (h < best_height)
2 |     best_height = h;
```

☐ B.

```
1 | if (h > best_height)
2 |     best_height = h;
```

☐ C.

```
1 | if (h > target)
2 |     best_height = h;
```

☐ D.

```
1 | if (h >= best_height)
2 |     best_height = h;
```

2 判断题（每题 2 分，共 20 分）

题号	1	2	3	4	5	6	7	8	9	10
答案	✓	✓	×	×	×	✓	×	✓	×	×

第 1 题 C++ 程序执行 `cin` 或 `scanf` 语句时通常是使用键盘等输入设备来输入某个变量的值。

第 2 题 执行 C++ 语句 `cout << ( 5 && 2 );` 将输出 1 。

第 3 题 C++ 表达式 `2 * - '2' / 4` 的值是 -1 。

第 4 题 如下 C++ 代码执行后的输出是 10 。

```
1 | int a;
2 | a = 5;
3 | a == +5;
4 | cout << a;
```

第 5 题 执行 C++ 代码 `cout << (not not 5)` 将输出 5 。

第 6 题 下面 C++ 代码执行后将输出 369 。

```
1 | for (int i = 1; i < 10; i++)
2 |     if (not(i % 3))
3 |         cout << i;
```

第 7 题 N 是整数，执行如下 C++ 代码，将能输出 0 到 N 之间(包含 N)所有数之和。

```
1 | int N, tot, i;
2 | cin >> N;
3 | tot = 0;
4 | for (i = 0; i < N + 1; i++)
5 |     tot += i;
6 | cout << tot;
```

第 8 题 如下 C++ 代码执行后，输出值为 0 。

```

1  int cnt, i, j;
2  cnt = 0;
3  for (i = 0; i < 10; i++) {
4      continue;
5      for (j = 0; j < i; j++)
6          cnt += 1;
7  }
8  cout << i * cnt;

```

第9题 求解鸡兔同笼。在如下 C++ 代码中，`tou` 代表鸡和兔的头总数，`jiao` 代表鸡和兔的脚总数。先后两个输入均为正整数。该程序能求解鸡兔同笼。

```

1  int tou, jiao, tu_tou, ji_tou;
2
3  cout << "请输入鸡和兔的头总数: ";
4  cin >> tou;
5  cout << "请输入鸡和兔的脚总数: ";
6  cin >> jiao;
7
8  for (tu_tou = 0; tu_tou < tou + 1; tu_tou++) {
9      ji_tou = tou - tu_tou;
10     if (tu_tou * 4 + ji_tou * 2 == jiao)
11         printf("兔的数量=%d,鸡的数量=%d\n", tu_tou, ji_tou);
12 }
13 if (tu_tou == tou)
14     printf("无解\n");

```

第10题 如果 `N` 和 `M` 均为整数，下列 C++ 代码则可以判断正整数 `N` 是 `M` 的平方。比如 9 是 3 的平方。

```

1  int N, M;
2
3  cout << "请输入正整数N: ";
4  cin >> N;
5  cout << "请输入整数M: ";
6  cin >> M;
7
8  if (int(sqrt(N)) == abs(M))
9      printf("%d是%d的平方\n", N, M);

```

3 编程题（每题 25 分，共 50 分）

3.1 编程题 1

- 试题名称：直角三角形
- 时间限制：1.0 s
- 内存限制：512.0 MB

3.1.1 题目描述

小双刚学会勾股定理，得知直角三角形的斜边的平方等于两直角边的平方和。

小双创造了直角三角形数列：数列前两项由小双指定，从第三项开始，数列中的每个数字的平方等于前两项的平方和。

例如，第一项是 3.0，第二项是 4.0，则第三项是 $\sqrt{3.0^2 + 4.0^2} = 5.0$ 。

其中 $\sqrt{x}$ 表示 x 的算术平方根，可以使用 C/C++/Python 标准库中对应函数计算。

小双给定了数列的前两项，并给定一个数列上限，想让你算算数列的第几项会超过小双给定的数列上限？

3.1.2 输入格式

输入三行，第一行一个浮点数是数列第一项，第二行一个浮点数是数列第二项，第三行一个浮点数表示数列上限。

3.1.3 输出格式

输出一个正整数，表示第几项开始会超过小双给定的数列上限。

3.1.4 样例

3.1.5 输入样例 1

```
1 3.0
2 4.0
3 10.0
```

3.1.6 输出样例 1

```
1 6
```

3.1.7 样例解释 1

数列依次为：

- 3.0,
- 4.0,
- $\sqrt{3.0^2 + 4.0^2} = 5.0$,
- $\sqrt{4.0^2 + 5.0^2} = 6.403124 \dots$,
- $\sqrt{5.0^2 + 6.403124^2} = 8.124038 \dots$,
- $\sqrt{6.403124^2 + 8.124038^2} = 10.344080 \dots$,

因此第 6 项会超过上限。

3.1.8 数据范围

输入均为不超过 10000 的正数。

题目保证给定第一项小于第二项。

3.1.9 参考程序

```
1 #include <iostream>
2 #include <cmath>
3 using namespace std;
4
5 int main() {
6     double a, b, U;
7     cin >> a >> b >> U;
8     int ans = 1;
9     if (a <= U) ans++;
10    while (b <= U) {
11        ans++;
12        double t = sqrt(a * a + b * b);
13        a = b, b = t;
14    }
15    cout << ans << endl;
16    return 0;
17 }
```

3.2 编程题 2

- 试题名称：字符变换
- 时间限制：1.0 s
- 内存限制：512.0 MB

3.2.1 题目描述

小明有一个 8×16 的画布（8 行 16 列），画布中的格子从上到下从左到右编号，左上的编号为 1，第一排最后一个的编号为 16，最后一排第一个编号为 113，右下为 128。

初始画布中的每个格子都是 `*`，随后小明会给定三个正整数，小明会将对应编号的格子变为该正整数对应的 *ASCII* 编码字符。

最后请帮小明输出整个画布。

3.2.2 输入格式

输入包含三行，每行一个正整数，表示需要替换的画布格子编号。

3.2.3 输出格式

输出 8×16 的画布。

3.2.4 样例

3.2.5 输入样例 1

```
1 48
2 65
3 97
```

3.2.6 输出样例 1

```
1 *****
2 *****
3 *****0
4 *****
5 A*****
6 *****
7 a*****
8 *****
```

3.2.7 数据范围

输入范围是 48 到 57，65 到 90，以及 97 到 122（即 *ASCII* 编码的 `0` 到 `9`，`A` 到 `Z`，以及 `a` 到 `z`）。

3.2.8 参考程序

```
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int a, b, c, p = 0;
6      cin >> a >> b >> c;
7      for (int i = 1; i <= 8; ++i) {
8          for (int j = 1; j <= 16; ++j) {
9              ++p;
10             if (p == a)
11                 cout << (char)a;
12             else if (p == b)
13                 cout << (char)b;
14             else if (p == c)
15                 cout << (char)c;
16             else
17                 cout << '*';
18         }
19         cout << '\n';
20     }
21     return 0;
22 }
```