

初中组 C++试卷

一、单选题（每小题 3 分，共计 30 分）

1. 计算机存储基本单位是（ ）。

- A.bit
- B.Byte
- C.KB
- D.MB

2. 十进制数 20252025 转换成八进制数字是（ ）。

- A.115202571
- B.151202571
- C.125202571
- D.112025171

3. 一个自然数在十进制下有 n 位，那么它在二进制下的位数与（ ）最接近。

- A. $5n$
- B. $n\log_2(10)$
- C. $10\log_2(10)$
- D. $10\log_2(n)$

4. 已知 2025 年 5 月 1 日是周四，那么 2015 年 5 月 1 日是周几（ ）？

- A.周二
- B.周三
- C.周四
- D.周五

5. 一棵树的前序遍历为 ABEDFCGH，中序遍历为 DFEBAGCH，此树的后序遍历为（ ）。

- A.FEBGDHCA
- B.FEDBGHCA
- C.FDEBGHCA
- D.FEDBGCHA

6. 一个简单无向图共有 16 条边，每个点的度数均为 4，那么这个图一共有（ ）个点。

- A.4
- B.8
- C.12
- D.16

7. 不包含数字 1 的三位数（不含前导零）有（ ）个。

- A.900
- B.648
- C.729
- D.836

8. 下列各种说法中，正确的是（ ）。

- A.计算机通过 IP 地址来访问网络中的其他设备
- B.家里的千兆宽带的带宽为 1000Mbps，下载最高速度是 1000MB/S
- C.常见的计算机网络操作系统包括 Unix、Linux、Android 等
- D.开源软件可以自由地进行复制、修改和传播，不需要得到授权

9. 设 $A=true, B=false, C=true, D=false$ ，以下逻辑运算表达式值为真的是（ ）。

- A. $(B \vee C \vee D) \wedge D \wedge A$
- B. $((\neg A \wedge B) \vee C) \wedge \neg D$
- C. $(A \wedge B) \vee (C \wedge D \vee \neg A)$
- D. $A \wedge (D \vee \neg C) \wedge B$

10. 1582 年以来公历的闰年规则是：

- 普通闰年：公历年份是 4 的倍数，且不是 100 的倍数的，为闰年（如 2004 年、2020 年等就是闰年）。
- 世纪闰年：公历年份是整百数的，必须是 400 的倍数才是闰年（如 1900 年不是闰年，2000 年是闰年）。

• 已知年份 x 满足 $x > 1582$ ，以下哪个逻辑表达式可以正确判定闰年（ ）。

- A. $x \% 4 == 0 \&\& x \% 100 != 0$
- B. $x \% 4 == 0 \parallel (x \% 400 == 0 \&\& x \% 100 != 0)$
- C. $x \% 4 == 0 \&\& (x \% 100 != 0 \parallel x \% 400 == 0)$
- D. $x \% 4 == 0 \&\& x \% 400 == 0 \&\& x \% 100 != 0$

11. 有以下代码，调用 `fun(5,2)` 会返回的结果是（ ）。

```
```cpp
int fun(int n, int m) {
 if (m == 0 || m == n) return 1;
 else return fun(n - 1, m - 1) + fun(n - 1, m);
}
```
```

- A.7
- B.8
- C.9
- D.10

12. 表达式 $(a+b) * c + d * (e+f)$ 的后缀形式是（ ）。

- A.`abc * def * +`

- B.abc * +def+
- C.ab+c * def+ * +
- D.ab+c * +def+

13. 在 C++中定义如下数组 `double a[10][100]` (`double` 为双精度浮点数类型), 存储容量约为 ()。

- A.2MB
- B.4KB
- C.8KB
- D.16MB

14. 下列是一个合法的电子邮件的是 ()。

- A.
- B.
- C.ccf@ccf.com
- D.CCF@

15 双向链表中有两个指针域 `link` 和 `rlink`, 分别指向该结点的前驱和后继。设 `p` 指向链表中的一个结点, 它的左右结点均非空。现要求删除结点 `p`, 则下面语句序列中错误的是 ()。

- A.`p->rlink->link=p->rlink;p->link->rlink=p->link;delete(p)`
- B.`p->link->rlink=p->rlink;p->rlink->link=p->link;delete(p)`
- C.`p->rlink->link=p->link;p->rlink->rlink=p->rlink;delete(p)`
- D.`p->link->rlink=p->rlink;p->rlink->link=p->link;delete(p)`

二、阅读程序 (程序输入不超过数组或字符串定义的范围; 除特殊说明外, 判断题 2 分, 选择题 3 分, 共计 40 分)

(1)

```
```cpp
#include<bits/stdc++.h>
using namespace std;
int a[105], n, x, ans = 0;
int main () {
 cin >> n;
 for(int i = 1; i <= n; i++){
 cin >> x;
 a[x]++;
 }
 for(int i = 0; i <= 100; i++){
 ans += a[i] % 2;
 }
}
```

```

 a[i+1] += a[i] / 2;

 }
 cout << ans << endl;
 return 0;
}
```

```

其中保证输入的 $1 \leq n, x \leq 100$

16. 最终输出的结果可能大于 n ()。

- A.正确
- B.错误

17. 当 $n=49$ 时，存在一种输入使得输出的结果为 1 ()。

- A.正确
- B.错误

18. 当 $n=50$ 时，存在一种输入使得输出的结果为 1 ()。

- A.正确
- B.错误

19. 若输入为 5 1 1 2 3 3，则输出为 ()。

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

20. 若输入 $n=20$ ，接下来所有数字均为 98，则输出为 ()。

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

(2)

```

```cpp
#include<bits/stdc++.h>
using namespace std;
const int mod = 1000000007;
int prime[1000010], tot, cnt[1000010], n;
bool vis[1000010];
long long ans = 1;
void init(){

```

```

 for(int i = 2; i <= n; i++){
 if (!vis[i])
 prime[++tot] = i;
 for(int j = 1; j <= tot; j++){
 if (prime[j] * i > n) break;
 vis[prime[j] * i] = 1;
 if (i % prime[j] == 0) break;
 }
 }
}
int main() {
 cin >> n;
 init();
 for(int i = 1; i <= tot; i++)
 for(int j = prime[i]; j <= n; j += prime[i])
 for(int k = j; k % prime[i] == 0; k /= prime[i])
 cnt[i]++;
 for(int i = 1; i <= tot; i++)
 ans = (ans * 1LL * (cnt[i] * 2 + 1) % mod) % mod;
 cout << ans << endl;
 return 0;
}
...

```

其中保证输入的  $n$  是不超过 1000000 的正整数

21. 将第 8 行的`int i=2;`改为`int i=1;`，程序输出不变（ ）。

- A.正确
- B.错误

22. 将第 12 行删除，程序会运行出错（ ）。

- A.正确
- B.错误

23. 将第 14 行删除，程序输出不变（ ）。

- A.正确
- B.错误

24. 将第 22 行的`int j=prime[i];`改为`int j=0;`，程序出现错误（运行错误或者答案错误）（ ）。

- A.正确
- B.错误

25. 如果输入的  $n$  为 2，输出为（）。

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

26. 该代码中`init()`函数的时间复杂度为（）。

- A. $O(\log n)$
- B. $O(\sqrt{n})$
- C. $O(n)$
- D. $O(n \log n)$

(3)

```
```cpp
#include<iostream>
using namespace std;
int a, b, x, y;
int main(){
    cin >> a >> b;
    while(1){
        x = a ^ b;
        y = a & b;
        if(y == 0) break;
        a = x;
        b = y << 1;
    }
    cout << x << endl;
    return 0;
}
```
```

其中输入保证  $0 \leq a, b \leq 10^9$

27. 第 11 行的`<<`和第 13 行的`<<`意义相同（）。

- A.正确
- B.错误

28. 第 7 行计算得到的  $x$  每次比上一次计算得到的大 ( )。
- A.正确
  - B.错误
29. 存在一组注释所示范围内的  $a, b$  使程序会陷入死循环 ( )。
- A.正确
  - B.错误
30. 如果输入的  $a=155, b=229$ , 那么第 10 行的代码会执行 ( ) 次。
- A.7
  - B.8
  - C.9
  - D.10
31. 设  $n=\max(a, b)$ , 最坏情况下, 此程序的时间复杂度是 ( )。
- A. $O(n)$
  - B. $O(\log n)$
  - C. $O(n \log n)$
  - D. $O(\sqrt{n})$

三、完善程序 (单选题, 每小题 3 分, 共计 30 分)

(1) (M拱门数组)

对于一个长度为  $n$  的非负整数数组  $a$ , 只要存在一组  $1 < x < y < z < n$  满足以下条件

- 1、 $a_1 < a_2 < \dots < a_x$
- 2、 $a_x > a_{x+1} > a_{x+2} > \dots > a_y$
- 3、 $a_y < a_{y+1} < a_{y+2} < \dots < a_z$
- 4、 $a_z > a_{z+1} > a_{z+2} > \dots > a_n$

(1)(M 拱门数组)

对于一个长度为  $n$  的非负整数数组  $a$ , 只要存在一组  $1 < x < y < z < n$  满足以下条件, 那么这个数组就是一个 M 拱门数组, 现在请完善以下程序来判断一个数组是不是 M 拱门数组。

- 1、 $a_1 < a_2 < \dots < a_x$
- 2、 $a_x >$

```
```cpp
#include<bits/stdc++.h>
using namespace std;
const int N = 1000100;
int T, n, tot, a[N];
```

```

int main(){
    cin >> n;
    for(int i = 1; i <= n; ++i){
        cin >> a[i];
    }
    int tot = 0;
    int flag = true;
    if (_____(1)_____) {
        flag = false;
    }
    for(int i = 1; i <= n - 1; ++i){
        if (_____(2)_____) {
            flag = false;
        }
    }
    for (int i = 2; i <= _____(3)_____; ++i){
        if (a[i - 1] < a[i] && a[i] > a[i + 1]){
            tot++;
        } if (a[i - 1] > a[i] && a[i] < a[i + 1]){
            _____(4)_____
        }
    }

    if(_____(5)_____ && flag){
        printf("Yes!\n");
    } else {
        printf("No!\n");
    }
    return 0;
}
```

```

32. (1)处应填 ( ) 。

- A.  $a[1] \geq a[2]$
- B.  $a[n-1] \leq a[n]$
- C.  $a[1] \geq a[2] \parallel a[n-1] \leq a[n]$
- D.  $a[1] \geq a[2] \&\& a[n-1] \leq a[n]$

33. (2)处应填 ( ) 。

- A.  $a[i] == a[i + 1]$
- B.  $a[i] == a[i - 1]$
- C.  $a[i] > a[i + 1]$
- D.  $a[i] < a[i + 1]$

34. (3)处应填 ( ) 。

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

35. (4)处应填 ( ) 。

- A.flag=false
- B.flag=true
- C.tot++
- D.tot--

36. (5)处应填 ( ) 。

- A.tot<=3
- B.tot==3
- C.tot>=3
- D.tot!=0

(2)(密码匹配)

给出两个字符串密码 A,B (保证长度相同)，现在需要匹配 A,B 两个密码中有多少位是一样的，即有多少个  $i$  (满足  $A[i]==B[i]$ )，但是 A,B 的长度很大，所以输入给出两个字符串 s 和 t，将 s 重复拼接 n 次得到 A，将 t 重复拼接 m 次得到 B，完成以下程序来完成这道题。

```
```cpp
#include<bits/stdc++.h>
using namespace std;
long long n, m, cnt[30], ans;
int ls, lt, c;
char s[1000100], t[1000100];
int gcd(int a, int b){
    if (!b) return a;
    return _____(1)_____;
}
int main(){
    cin >> n >> m;
    cin >> s >> t;
    ls = strlen(s);
    lt = strlen(t);
    c = gcd(ls, lt);
    for (int i = 0; i < c; i++){
        memset(cnt, 0, sizeof(cnt));
        for (int j = 0; j < _____(2)_____; j++){
```

```

        cnt[_____(3)_____]++;
    }
    for (int j = 0; j < lt / c; j++){
        ans += cnt[_____(4)_____ - 'a'];
    }
}
ans *= _____(5)_____;
printf("%lld\n", ans);
return 0;
}
'''

```

37. (21)处应填 () 。

- A.gcd(a,b)
- B.gcd(b,a%b)
- C.gcd(a%b,b)
- D.gcd(a%b,a)

38. (22)处应填 () 。

- A.ls
- B.c
- C.ls/c
- D.n

39. (3)处应填 () 。

- A.s[i * c+j]
- B.s[j * c+i]
- C.s[i * c+j] - 'a'
- D.s[j * c+i] - 'a'

40. (4)处应填 () 。

- A.t[i * c+j]
- B.t[j * c+i]
- C.t[i * c+j] - 'a'
- D.t[j * c+i] - 'a'

41. (5)处应填 () 。

- A.n * c/lt
- B.ls * lt/c
- C.n * m/ls

- $D \cdot n \cdot m/c$