C++ do-while 循环 课后作业

姓名: _____

一、 填空题

1. `do-while` 循环的核心思想是“先 _____，后 _____”。
2. 与 `while` 循环不同，`do-while` 循环的循环体代码保证至少会被执行 _____ 次。
3. `do-while` 循环的语法结构在末尾的 `while(条件)` 之后必须有一个 _____。
4. 在猜数字游戏中，因为玩家无论如何都需要先输入一次猜测的数字，所以使用 _____ 循环比 `while` 循环更合适。
5. 在 C++ 中，为了生成每次运行程序时都不同的随机数，我们需要使用 `time(0)` 函数来获取当前的 _____，并通过 `srand()` 函数设置随机数 _____，为了使用这两个函数工具，需要包含 `<ctime>` 和 `<cstdlib>` 两个头文件。

二、 单项选择题

(　) 1. 关于 `do-while` 和 `while` 循环，下列说法正确的是?

A. 两者没有任何区别
B. `while` 循环的循环体至少执行一次
C. `do-while` 循环的循环体至少执行一次
D. `do-while` 循环先判断条件再执行

(　) 2. 观察下面的代码，它会输出多少个 "Hello"?

```
int i = 10;
do {
    cout << "Hello" << endl;
    i++;
} while (i < 10);
```

A. 0 个　B. 1 个　C. 10 个　D. 无限个

(　) 3. 哪个场景最适合使用 `do-while` 循环?

A. 遍历一个已知长度的数组
B. 计算 1 到 100 的累加和
C. 程序启动后，先显示菜单让用户选择，如果选择无效则重新显示
D. 从 100 倒数到 1

(　) 4. 要生成一个 `[1, 50]` 范围内的随机整数，正确的表达式是?

A. `rand() % 49`　　B. `rand() % 50`
C. `rand() % 50 + 1`　　D. `rand() % 51`

(　) 5. 在猜数字游戏中，`do-while` 循环的循环条件通常是?

A. `guess == ans`　　B. `guess != ans`
C. `guess > ans`　　D. `guess < ans`

(　) 6. 分析以下代码，最终输出结果是什么?

```
int n = 5;
do {
    cout << n << " ";
    n--;
} while (n > 5);
```

A. 什么都不输出　B. 5 C. 5 4 3 2 1　D. 导致死循环

(　) 7. 关于 `srand(time(0));` 这行代码，下列描述错误的是?

A. 它的作用是初始化随机数种子 B. `time(0)` 提供一个不断变化的值

C. 它应该被放在 `do-while` 循环的内部，以保证每次循环都足够随机

D. 它能确保程序每次运行时，生成的随机数序列都不同

(　) 8. `do-while` 循环的条件判断发生在哪个时间点?

A. 每次循环体执行之前

B. 每次循环体执行之后

C. 仅在整个循环开始前判断一次

D. 仅在循环体执行到一半时

(　) 9. 以下哪个 `do-while` 的语法是正确的?

A. `do while(x<5) { ... }`

B. `do { ... } while x<5;`

C. `do { ... } while(x<5)`

D. `do { ... } while(x<5);`

(　) 10. 如果有一个程序需要反复验证用户输入的密码，直到输入正确为止，最合适的循环结构是?

A. `if-else` 结构

B. `while` 循环，并在循环外先获取一次输入

C. `do-while` 循环

D. 选项 B 和 C 都可以，但 C 更简洁优雅

三、 判断题 (对的打“√”，错的打“×”)

(　) 1. `do-while` 循环的循环条件即使初始就为 `false`，循环体也依然会执行一次。

(　) 2. 在任何情况下，一个 `while` 循环都可以被等价地改写为 `do-while` 循环。

(　) 3. `rand() % 100` 生成的随机数范围是 `[0, 100]`，即包含0和100。

(　) 4. 在 `do { ... } while(条件表达式);` 结构中，在执行完循环体后，`条件表达式` 变为 `false` 时循环终止。

(　) 5. 因为 `do-while` 循环体至少执行一次，所以它永远不会导致“一次都不执行”的情况。

(　) 6. `srand(time(0));` 这行代码的作用是生成一个随机数。

(　) 7. `do-while` 循环适合于需要先获取用户输入再进行判断的逻辑。

(　) 8. 如果忘记了 `do-while` 语句末尾的分号，程序将无法通过编译。

(　) 9. 在一个 `do-while` 循环体内，必须有能够影响到循环条件的语句，否则可能导致死循环。

(　) 10. `while(guess != ans)` 和 `while(!(guess == ans))` 这两个循环条件是完全等价的。