C++primer 第五版 第五章 答案

5.1 答：最简单的是空语句，空语句中只含有一个单独的分号；

```
    while（cin>>s&&s!=sought）
        ;//空语句
```

5.2 答：块是用花括号括起来的语句和声明的序列，其实就是复合语句。在 while 循环中就用到块。

5.3 答：while（int sum=0，val=1；val<=10;sum+=val,++val）

5.4 答：（a）itr 没有初值，应该在前面加上 auto itr=s.begin()；（b）应是用 find 函数查找某个单词，但是 word 的没有初值，应该 while 之前给 word 赋值；

5.5 答：#include <iostream>

```
#include <string>
#include <vector>
using namespace std;
int main()
{
    const vector<string> scores = { "F", "D", "C", "B", "A", "A++" };
    vector<unsigned> grades;
    unsigned grade;
    while (cin >> grade)
        grades.push_back(grade);
    for (vector<unsigned>::const_iterator it = grades.begin();
        it != grades.end(); ++it)
    {
        cout << *it << " ";
        string lettergrade;
        if (*it < 60)
            lettergrade = scores[0];
        else
        {
            lettergrade = scores[(*it - 50) / 10];
            if (*it != 100)
            {
                if (*it % 10 > 7)
                    lettergrade += '+';
                else if (*it % 10 < 3)
                    lettergrade += '-';
            }
        }
        cout << lettergrade << endl;
    }
}
```

5.6 答：#include <iostream>

```
#include <string>
#include <vector>
using namespace std;
int main()
{
    const vector<string> scores = { "F", "D", "C", "B", "A", "A++" };
    vector<unsigned> grades;
    unsigned grade;
    while (cin >> grade)
        grades.push_back(grade);
    for (vector<unsigned>::const_iterator it = grades.begin();
        it != grades.end(); ++it)
    {
        cout << *it << " ";
        string lettergrade;
        if (*it < 60)
            lettergrade = scores[0];
        else
        {
            lettergrade = scores[(*it - 50) / 10];
            if (*it != 100)
{
            (*it%10>7) ? (lettergrade += '+') : (*it % 10<3) ? (lettergrade += '-') : (lettergrade);
}
        }
        cout << lettergrade << endl;
    }
}
```

5.7 答：（a）少一个分号；（b）把 if 后面的语句用花括号括起来；（c）在循环体前定义 int ival；（d）改成 if（ival==0）；

5.8 答：悬垂 else 是指在一个语句包含的 if 子语句多余 else 子语句的时候，多余的 else 应该与哪一个 if 匹配。

5.9 答：

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    unsigned aCnt = 0, eCnt = 0, iCnt = 0, oCnt = 0, uCnt = 0;
    char ch;
    while (cin >> ch)
    {
        if (ch == 'a')
            aCnt++;
        if (ch == 'e')
```

```
            eCnt++;
        if (ch == 'i')
            iCnt++;
        if (ch == 'o')
            iCnt++;
        if (ch == 'u')
            uCnt++;
    }
    cout << "Number of vowel a: \t" << aCnt << '\n'
        << "Number of vowel e: \t" << eCnt << '\n'
        << "Number of vowel i: \t" << iCnt << '\n'
        << "Number of vowel o: \t" << oCnt << '\n'
        << "Number of vowel u: \t" << uCnt << endl;

    return 0;
}
```

```
5.10 答：#include <iostream>
#include <string>
using namespace std;
int main()
{
    unsigned aCnt = 0, eCnt = 0, iCnt = 0, oCnt = 0, uCnt = 0;
    char ch;
    while (cin >> ch) {
        switch (ch) {
        case 'A':
        case 'a':
            ++aCnt;
            break;
        case 'E':
        case 'e':
            ++eCnt;
            break;
        case 'I':
        case 'i':
            ++iCnt;
            break;
        case 'O':
        case 'o':
            ++oCnt;
            break;
        case 'U':
        case 'u':
            ++uCnt;
            break;
        }
    }
    cout << "Number of vowel a: \t" << aCnt << '\n'
        << "Number of vowel e: \t" << eCnt << '\n'
        << "Number of vowel i: \t" << iCnt << '\n'
        << "Number of vowel o: \t" << oCnt << '\n'
        << "Number of vowel u: \t" << uCnt << endl;
    return 0;
}
5.11 答：#include <iostream>
#include <string>
using namespace std;
int main()
{
    unsigned aCnt = 0, eCnt = 0, iCnt = 0, oCnt = 0, uCnt = 0;
    unsigned spaceCnt = 0, tabCnt = 0, newlineCnt = 0;
    char ch;
    while (cin >> ch) {
        switch (ch) {
        case 'A':
        case 'a':
            ++aCnt;
            break;
        case 'E':
        case 'e':
            ++eCnt;
            break;
        case 'I':
        case 'i':
            ++iCnt;
            break;
        case 'O':
        case 'o':
            ++oCnt;
            break;
        case 'U':
        case 'u':
            ++uCnt;
            break;
        case ' ':
            ++spaceCnt;
            break;
        case '\t':
            ++tabCnt;
            break;
        case '\n':
            ++newlineCnt;
            break;
        }
    }
    cout << "Number of vowel a: \t" << aCnt << '\n'
        << "Number of vowel e: \t" << eCnt << '\n'
        << "Number of vowel i: \t" << iCnt << '\n'
        << "Number of vowel o: \t" << oCnt << '\n'
        << "Number of vowel u: \t" << uCnt << endl
        << "Number of vowel space: \t" << spaceCnt << endl
        << "Number of vowel tab: \t" << tabCnt << endl
        << "Number of vowel newline: \t" << newlineCnt << endl;
    return 0;
}
```

5.12 答：#include <iostream>

```
#include <string>
using namespace std;
int main()
{
    char ch, c='\0';
    unsigned ffCnt = 0, flCnt = 0, fiCnt = 0;
    while (cin>>ch)
    {
        if (c == 'f')
        {
            switch (ch)
            {
            case 'f':
                ffCnt++;
                break;
            case 'l':
                flCnt++;
                break;
            case 'i':
                fiCnt++;
                break;
            default:
                break;
            }
        }
        c = ch;
    }
    cout << ffCnt << endl << flCnt << endl << fiCnt << endl;
    return 0;
}
```

5.13 答：（a）缺少 break 语句；（b）在 case1 语句之后和 default 预计之前定义了变量，应该把 int ix；在 switch 之前就可以了；（c）case 语句之后出现了多个常量，应该用分别 case 1：case 2：....这样写；（d）case 语句应该接常量或常量表达式，不应该是变量，ival、jval、kval 都是变量。

```
5.14 答：#include <iostream>
#include <string>
using namespace std;
int main()
{
    string  BeforeWord,NowWord  ,ResultsWord;//用于保存上一次输入,当前输入和最大的结果
    int cnt = 0, MaxCnt = 1;//计数和保存最大数
    while (cin >> NowWord)//输入
    {
        if (NowWord == BeforeWord)//以上一个输入相同
        {
            ++cnt;
        }
        else//以上一个输入不同
        {
            if (cnt>MaxCnt)
            {
                MaxCnt = cnt;
                ResultsWord = BeforeWord;
            }
            cnt = 1;
        }
        BeforeWord = NowWord;
    }
    if (MaxCnt != 1)//保证是在有重复的情况之下
    {
        cout << ResultsWord << endl << MaxCnt;
    }
    return 0;
}
```

5.15 答：（a）ix 不等于 sz 执行循环体，循环体中应该有跳出循环的语句（break 或 return）来执行后面的 if 语句，不然 if 条件永远为假；（b）ix 应该赋初值，当 ix 不等于 sz 的时候执行循环体；（c）当 sz 不等于 0 的时候循环变成一个无限循环或者说是一个死循环。

```
5.16 答：const size_t sz = 10;
    int a[sz];
    size_t index =0;

    for (; index != sz; ++index)
    {
        //操作元素
    }

    while (index != sz)
    {
        //操作元素
        ++index;
    }

    do{
        //操作元素
        ++index;
    } while (index != sz);
```

}//每种循环都有自己各自的特点。功能上都是等价的，根据自己的爱好或者程序的需要选择就行。

5.17 答：#include <iostream>

```
#include<vector>
#include<string>
using namespace std;
bool check(vector<int> ivec1, vector<int> ivec2, size_t size);
int main()
{
    vector<int> ivec1, ivec2;
    int i = 0;
    cout << "input ivec2: ";
    while (cin >> i)
        ivec1.push_back(i);
    cin.clear();
    cout << "input ivec2: ";
    while (cin >> i)
        ivec2.push_back(i);
    size_t size_1 = ivec1.size(), size_2 = ivec2.size();
    bool result = size_1>size_2 ? check(ivec1, ivec2, size_2) : check(ivec1, ivec2, size_1);
    cout << boolalpha << result << endl;
}

bool check(vector<int> ivec1, vector<int> ivec2, size_t size)
{
    for (size_t i = 0; i < size; ++i)
    {
        if (ivec1[i] != ivec2[i])
            return false;
    }
    return true;
}
```

5.18 答：（a）do 与 while 之间缺少花括号；（b）不允许在条件部分定义变量；（c）正确；

```
5.19 答：#include <iostream>
#include<string>
using namespace std;
int main()
{
    do{
        cout << "输入两个 string：";
        string v1,v2;
        cin >> v1 >> v2;
        if (v1.size() > v2.size())
            cout << v2 << endl;
        else
            cout << v1 << endl;

    } while (cin);
}
```

```
5.20 答：#include <iostream>
#include<string>
using namespace std;
int main()
{
    string s1, s2;
    while (cin >> s2)
    {
        if (s2 == s1)
            break;
        else
            s1=s2;
    }
    if ((!s1.empty()) && s1 == s2 )
        cout << "重复的单词是： " << s2 << endl;
    else
        cout << "没有重复的单词" << endl;
    return 0;
}
```

```
5.21 答：#include <iostream>
#include<string>
using namespace std;
```

```
int main()
{
    string s1, s2;
    while (cin >> s2)
    {
        if (s2 == s1&& isupper(s2[0]))
            break;
        else
            s1=s2;
    }
    if ((!s1.empty()) && s1 == s2 )
        cout << "重复的单词是：" << s2 << endl;
    else
        cout << "没有重复的单词" << endl;
    return 0;
}
```

5.22 答：

```
do{
    int sz = get_size();
} while (sz < 0);
```

5.23 答：

```
#include <iostream>
#include<string>
using namespace std;
int main()
{
    int ival1, ival2;
    cin >> ival1 >> ival2;
    cout << ival1 + ival2 << endl;
}
```

5.24 答：

```
#include <iostream>
#include<string>
using namespace std;
int main()
{
    int ival1, ival2;
    cin >> ival1 >> ival2;
    if (ival2 == 0)
        throw runtime_error("第二个数为 0");
    cout << ival1 + ival2 << endl;
}
```

5.25 答：

```
#include <iostream>
#include<string>
using namespace std;
int main()
{
    int ival1, ival2;
    while (cin >> ival1 >> ival2)
    {
        try{
            if (ival2==0)
                throw runtime_error("第二个数为 0");
            cout << ival1 + ival2 << endl;
        }
        catch (runtime_error err){
            cout << err.what()
                << "\nTry Again?    Enter y or n" << endl;
            char c;
            cin >> c;
            if (!cin || c == 'n')
                break;
        }
    }
}
```