FINALS SENTENCES C++

```cpp
string getWord(vector<string> d, char c, int cnt)
{
 int pos = 0;
 for (; d[pos][0] != c; ++pos)
 {
 };
 while (cnt)
 {
 stringstream ss;
 ss << d[pos];
 string temp;
 ss >> temp;
ss te p;
 temp = "";
 while (!ss.eof())
 {
 --cnt;
 ss >> temp;
 if (cnt == 0)
 return temp;
 temp = "";
 }
 }
 return "";
}
string generate_sentences(vector<string> dictionary, string str)
{
```

```cpp
char voc[] = "aeiou";

string ans = "";

int ok = 0;

switch (str[0])

{

case 'D':

ok = 1;

break;

case 'I':

ok = 2;

break;

case 'Q':

ok = 3;

break;

case 'E':

ok = 4;

break;

}

if (ok == 3)

ans += "What ";

int cn = 1, cc = 1, cv = 1, cj = 1, cb = 1, cp = 1;

for (int i = 1; i < str.size(); ++i)

{

if (str[i] == ' ')

{

switch (ok)

{

case 3:

ans += "? ";
```

```
            break;
            case 4:
            ans += "! ";
            break;
            default:
            ans += ". ";
            break;
            }
            ans += ' ';
            ++i;
            ok = 0;
            switch (str[i])
            {
            case 'D':
            ok = 1;
            break;
            case 'I':
            ok = 2;
            break;
            break;
            case 'Q':
            ok = 3;
            break;
            case 'E':
            ok = 4;
            break;
            }
            if (ok == 3)
            ans += "What ";
```

```
continue;

}

switch (str[i])

{

case 'T':

ans += "the ";

break;

case 'A':

ans += "a ";

break;

case 'N':

ans += getWord(dictionary, 'N', cn) + " ";

++cn;

break;

case 'C':

ans += getWord(dictionary, 'C', cc) + " ";

++cc;

break;

case 'V':

ans += getWord(dictionary, 'V', cv) + " ";

++cv;

break;

case 'J':

ans += getWord(dictionary, 'J', cj) + " ";

++cj;

break;

case 'B':

ans += getWord(dictionary, 'B', cb) + " ";

++cb;
```

```cpp
            break;
        case 'P':
            ans += getWord(dictionary, 'P', cp) + " ";
            ++cp;
            break;
        }
    }
    ans.erase(ans.begin() + ans.size() - 1);
    switch (ok)
    {
    case 3:
        ans += '?';
        break;
    case 4:
        ans += '!';
        break;
    default:
        ans += '.';
        break;
    }
    ans[0] = toupper(ans[0]);
    for (int i = 1; i < ans.size(); ++i)
    {
        if (ans[i] == ' ')
        {
            if (!isalpha(ans[i - 1]) || ans[i + 1] == ' ' || ans[i + 1] ==
            if (!isalpha(ans[i - 1]) || ans[i + 1] == . || ans[i + 1] ==
'!' || ans[i + 1] == '?')
            {
```

```cpp
            ans.erase(ans.begin() + i);

        }

    }

}

for (int i = 2; i < ans.size(); ++i)

{

    if (ans[i] == 'a' && ans[i + 1] == ' ' && strchr(voc, ans[i + 2]))

    {

        ans.insert(ans.begin() + i + 1, 'n');

    }

    if (ans[i - 2] == '.' || ans[i - 2] == '!' || ans[i - 2] == '?')

        ans[i] = toupper(ans[i]);

}

return ans;

}
```