Problem 1

Write a function

```
bool checkpass(const char* pass);
```

which takes a password (as a C-string, i.e., array of characters) and checks its validity. We assume that a valid password contains

1. at least 8 characters;
2. at least 6 different characters;
3. at least 1 digit;
4. at least 1 uppercase letter;
5. at least 1 lowercase letter;
6. at least 1 non-alphanumeric character (which is neither a letter nor a digit).

The function returns **true** if the password is correct; if it is not, **false** is returned, and before returning the function displays information about *all* causes of the failure. You can assume that all characters are plain ASCII characters with codes from the the range $[32, 126]$. [You may find useful to define some simple auxiliary functions.]

For example, the following program

download *CStringPass.cpp*

```
#include <iostream>

// ...

bool checkpass(const char* pass) {
    // ...
}

int main() {
    using std::cout; using std::endl;
    const char* passes[] =
        {
            "AbcDe93",  "A1b:A1b>",  "Ab:Acb<",
            "abc123><", "Zorro@123", nullptr
        };
    for (int i = 0; passes[i] != nullptr; ++i) {
        cout << "checking " << passes[i] << endl;
        if (checkpass(passes[i])) cout << "OK" << endl;
        cout << endl;
    }
}
```

should print something like

```
checking AbcDe93
Too short
No non-alphanumeric character

checking A1b:A1b>
Too few different characters

checking Ab:Acb<
Too short
Too few different characters
No digit

checking abc123><
No uppercase letter

checking Zorro@123
OK
```

Do not include any headers other than *iostream*!
