

```
/*  
Name: Leo Chi  
Team: STEMnArts  
Division: Junior  
Contest 3  
*/  
  
#include <iostream>  
  
using namespace std;  

```

```
        break;
case '4':
    tmp = "0100";
    break;
case '5':
    tmp = "0101";
    break;
case '6':
    tmp = "0110";
    break;
case '7':
    tmp = "0111";
    break;
case '8':
    tmp = "1000";
    break;
case '9':
    tmp = "1001";
    break;
case 'A':
case 'a':
    tmp = "1010";
    break;
case 'B':
case 'b':
```

```
        tmp = "1011";
        break;
case 'C':
case 'c':
        tmp = "1100";
        break;
case 'D':
case 'd':
        tmp = "1101";
        break;
case 'E':
case 'e':
        tmp = "1110";
        break;
case 'F':
case 'f':
        tmp = "1111";
        break;
}

return tmp;
}

// Makes string of equation from binary set
string Veitch(string bin)
```

```

{
    bin=" "+bin;

    string tmp = "potato";

    if (bin[1] == '1' && bin[2] == '1' && bin[3] == '1' && bin[4] == '1') {

        bin[1] = 0;

        bin[2] = 0;

        bin[3] = 0;

        bin[4] = 0;

        tmp = "B";

    }

    if (bin[5] == '1' && bin[6] == '1' && bin[7] == '1' && bin[8] == '1') {

        bin[5] = 0;

        bin[6] = 0;

        bin[7] = 0;

        bin[8] = 0;

        if (tmp == "potato") {

            tmp = "~B";

        } else {

            tmp += " + ~B";

        }

    }

    if (bin[1] == '1' && bin[2] == '1' && bin[5] == '1' && bin[6] == '1') {

        bin[1] = 0;

        bin[2] = 0;

        bin[5] = 0;

```

```
bin[6] = 0;
if (tmp == "potato") {
    tmp = "A";
} else {
    tmp += " + A";
}
}
if (bin[2] == '1' && bin[3] == '1' && bin[6] == '1' && bin[7] == '1') {
    bin[2] = 0;
    bin[3] = 0;
    bin[6] = 0;
    bin[7] = 0;
    if (tmp == "potato") {
        tmp = "C";
    } else {
        tmp += " + C";
    }
}
if (bin[3] == '1' && bin[4] == '1' && bin[7] == '1' && bin[8] == '1') {
    bin[3] = 0;
    bin[4] = 0;
    bin[7] = 0;
    bin[8] = 0;
    // cout<< "not A triggered" << endl;
```

```
if (tmp == "potato") {
    tmp = "~A";
} else {
    tmp += " + ~A";
}
}

if (bin[1] == '1' && bin[4] == '1' && bin[5] == '1' && bin[8] == '1') {
    bin[1] = 0;
    bin[4] = 0;
    bin[5] = 0;
    bin[8] = 0;
    if (tmp == "potato") {
        tmp = "~C";
    } else {
        tmp += " + ~C";
    }
}

if (bin[1] == '1' && bin[2] == '1') {
    bin[1] = 0;
    bin[2] = 0;
    if (tmp == "potato") {
        tmp = "AB";
    } else {
        tmp += " + AB";
    }
}
```

```
}  
  
if (bin[2] == '1' && bin[3] == '1') {  
    bin[2] = 0;  
  
    bin[3] = 0;  
  
    cout << "BC triggered" << endl;  
  
    if (tmp == "potato") {  
        tmp = "BC";  
  
    } else {  
        tmp += " + BC";  
  
    }  
  
}  
  
if (bin[3] == '1' && bin[4] == '1') {  
    bin[3] = 0;  
  
    bin[4] = 0;  
  
    if (tmp == "potato") {  
        tmp = "~AB";  
  
    } else {  
        tmp += " + ~AB";  
  
    }  
  
}  
  
if (bin[5] == '1' && bin[6] == '1') {  
    bin[5] = 0;  
  
    bin[6] = 0;  
  
    if (tmp == "potato") {
```

```
    tmp = "A~B";

} else {

    tmp += " + A~B";

}

}

if (bin[6] == '1' && bin[7] == '1') {

    bin[6] = 0;

    bin[7] = 0;

    if (tmp == "potato") {

        tmp = "~BC";

    } else {

        tmp += " + ~BC";

    }

}

if (bin[7] == '1' && bin[8] == '1') {

    bin[7] = 0;

    bin[8] = 0;

    if (tmp == "potato") {

        tmp = "~A~B";

    } else {

        tmp += " + ~A~B";

    }

}

if (bin[1] == '1' && bin[5] == '1') {

    bin[1] = 0;
```



```
bin[5] = 0;

if (tmp == "potato") {
    tmp = "A~C";
} else {
    tmp += " + A~C";
}
}

if (bin[2] == '1' && bin[6] == '1') {
    bin[2] = 0;
    bin[6] = 0;
    if (tmp == "potato") {
        tmp = "AC";
    } else {
        tmp += " + AC";
    }
}

if (bin[3] == '1' && bin[7] == '1') {
    bin[3] = 0;
    bin[7] = 0;
    if (tmp == "potato") {
        tmp = "~AC";
    } else {
        tmp += " + ~AC";
    }
}
```

```
}  
  
if (bin[4] == '1' && bin[8] == '1') {  
    bin[4] = 0;  
  
    bin[8] = 0;  
  
    if (tmp == "potato") {  
        tmp = "~A~C";  
  
    } else {  
        tmp += " + ~A~C";  
  
    }  
}  
  
if (bin[1] == '1' && bin[4] == '1') {  
    bin[1] = 0;  
  
    bin[4] = 0;  
  
    if (tmp == "potato") {  
        tmp = "B~C";  
  
    } else {  
        tmp += " + B~C";  
  
    }  
}  
  
if (bin[5] == '1' && bin[8] == '1') {  
    bin[5] = 0;  
  
    bin[8] = 0;  
  
    if (tmp == "potato") {  
        tmp = "~B~C";  
  
    } else {
```

```
    tmp += " + ~B~C";
}
}
if (bin[1] == '1') {
    bin[1] = 0;
    if (tmp == "potato") {
        tmp = "AB~C";
    } else {
        tmp += " + AB~C";
    }
}
if (bin[2] == '1') {
    bin[2] = 0;
    if (tmp == "potato") {
        tmp = "ABC";
    } else {
        tmp += " + ABC";
    }
}
if (bin[3] == '1') {
    bin[3] = 0;
    if (tmp == "potato") {
        tmp = "~ABC";
    } else {
```

```
    tmp += " + ~ABC";
}
}
if (bin[4] == '1') {
    bin[4] = 0;
    if (tmp == "potato") {
        tmp = "~AB~C";
    } else {
        tmp += " + ~AB~C";
    }
}
if (bin[5] == '1') {
    bin[5] = 0;
    if (tmp == "potato") {
        tmp = "A~B~C";
    } else {
        tmp += " + A~B~C";
    }
}
if (bin[6] == '1') {
    bin[6] = 0;
    if (tmp == "potato") {
        tmp = "A~BC";
    } else {
        tmp += " + A~BC";
    }
}
```

```
    }  
}  
if (bin[7] == '1') {  
    bin[7] = 0;  
    if (tmp == "potato") {  
        tmp = "~A~BC";  
    } else {  
        tmp += " + ~A~BC";  
    }  
}  
if (bin[8] == '1') {  
    bin[8] = 0;  
    if (tmp == "potato") {  
        tmp = "~A~B~C";  
    } else {  
        tmp += " + ~A~B~C";  
    }  
}  
return tmp;  
}
```

// Puts stuff together

```
string Assemble(string input)
```

```
{
```

```
string tmp = HexToBin(input[0]);  
tmp += HexToBin(input[1]);  
tmp = Veitch(tmp);  
return tmp;  
}
```

```
// main code
```

```
int main()  
{  
    string a, b, c, d, e, tmp;  
    cin >> a >> b >> c >> d >> e;  
    cout << Assemble(a) << endl;  
    cout << Assemble(b) << endl;  
    cout << Assemble(c) << endl;  
    cout << Assemble(d) << endl;  
    cout << Assemble(e) << endl;  
    return 0;  
}
```