

Плюсы:

- Наличие организации проекта в виде классов
- Не используется множественное наследование
- В коде содержатся «говорящие» названия классов, методов
- Именованние констант, методов, классов выдержано в едином стиле
- Не используются «магические» константы
- Не используются глобальные переменные
- Методы достаточно компактные, небольшого размера
- Отсутствуют закомментированные участки кода

Минусы:

- Встречаются длинные конструкции, что снижает читабельность кода

```
49     func session(_ session: MCSession, didReceive stream: InputStream, withName streamName: String, fromPeer peerID: MCPeerID) {  
50     }  
51  
52     func session(_ session: MCSession, didStartReceivingResourceWithName resourceName: String, fromPeer peerID: MCPeerID, with progress: Progress) {  
53     }  
54  
55     func session(_ session: MCSession, didFinishReceivingResourceWithName resourceName: String, fromPeer peerID: MCPeerID, at localURL: URL, withProgress progress: Progress) {  
56     }  
57 }
```

- Средняя комментируемость кода, комментарии присутствуют, но хотелось бы побольше
- Дублирование кода, возможно, эту конструкцию можно было бы доработать

```

if fields[0].player == x && fields[1].player == x && fields[2].player == x{
    winner = x
}else if fields[0].player == o && fields[1].player == o && fields[2].player == o{
    winner = o
}else if fields[3].player == x && fields[4].player == x && fields[5].player == x{
    winner = x
}else if fields[3].player == o && fields[4].player == o && fields[5].player == o{
    winner = o
}else if fields[6].player == x && fields[7].player == x && fields[8].player == x{
    winner = x
}else if fields[6].player == o && fields[7].player == o && fields[8].player == o{
    winner = o
}else if fields[0].player == x && fields[3].player == x && fields[6].player == x{
    winner = x
}else if fields[0].player == o && fields[3].player == o && fields[6].player == o{
    winner = o
}else if fields[1].player == x && fields[4].player == x && fields[7].player == x{
    winner = x
}else if fields[1].player == o && fields[4].player == o && fields[7].player == o{
    winner = o
}else if fields[2].player == x && fields[5].player == x && fields[8].player == x{
    winner = x
}else if fields[2].player == o && fields[5].player == o && fields[8].player == o{
    winner = o
}else if fields[0].player == x && fields[4].player == x && fields[8].player == x{
    winner = x
}else if fields[0].player == o && fields[4].player == o && fields[8].player == o{
    winner = o
}else if fields[2].player == x && fields[4].player == x && fields[6].player == x{

```

Оценка: 8/10