```
import SwiftUI
@main
struct ShultApp: App {
    var body: some Scene {
        WindowGroup {
            ContentView()
        }
    }
}
import Foundation
import SwiftUI
var numberArray: [Int] = generateArray(n:5)
func generateArray(n: Int) -> [Int] {
    var numberArray = Array(repeating: 0, count: n*n)
    for index in 1...n*n {
        numberArray[index-1] = index
    }
    return numberArray.shuffled()
}
//  ContentView.swift
//  Shult
//  Created by xu on 2022/2/28.
import SwiftUIstruct ContentView: View {

    @State var time = 0   // 计时器 对应 TimerView 视图

    @State var size = 5      // 每行显示数目，数据流向 GameView -> TableView

    @State var numberArray = generateArray(n: 5) // 生成宫格，数据流向 TableView

    @State var isStart = true
    @State var isPass = false
    var body: some View {
        GameView(time: $time, size: $size, numberArray: $numberArray, isPass: $isPass, isStart: $isStart)
            .sheet(isPresented: $isStart, content: {
                StartView(isStart: $isStart, time: $time, size: $size, numberArray: $numberArray)
                    .highPriorityGesture(DragGesture())
            }
            )
    }
}
struct StartView: View {
    @Binding var isStart: Bool

    @Binding var time: Int   // 计时器 对应 TimerView 视图
    @Binding var size: Int     // 每行显示数目，数据流向 TableView

    @Binding var numberArray: [Int] // 生成宫格，数据流向 TableView

    var body: some View {
        VStack{

            // 名称

            Spacer()
            Group {

                Text("极简舒尔特方格")

                    .font(.largeTitle)
            }
            Spacer()

            // 类型选择

//           TypeTableView()
            TypeTableView(size: $size, numberArray: $numberArray)
            Spacer()

            // 按钮区

            Group {

                // 准备开始

                Button(
                    action: {
                        self.isStart = false
//                       self.numberArray = generateArray(n: self.size)
                        self.time = 0
                    }
                ) {
                    Capsule()
                        .stroke(lineWidth: 3)
                        .frame(width: 120, height: 40)
                        .overlay{

                            Text("开始")

                                .font(.title)
                        }
                }
                .foregroundColor(.black)
            }
            .padding()
```

```
                Spacer()
            }
        }
}struct ContentView_Previews: PreviewProvider {
        static var previews: some View {
            ContentView()
        }
}
//    NextNumberView.swift
//    Shult
//    Created by xu on 2022/3/2.
//import SwiftUIstruct NextNumberView: View {
        @Binding var nextNumber: Int
        var body: some View {
            HStack{

                Text("下个数")

                Text("\(nextNumber)")
            }
            .font(.title2)
        }
}//struct NextNumberView_Previews: PreviewProvider {
//        static var previews: some View {
//                NextNumberView()
//        }
//}
//    GridView.swift
//    Shult
//    Created by xu on 2022/2/28.
//import SwiftUI
// UIScreen.main.bounds.width: get screen width
// UIScreen.main.bounds.heigth: get screen heigthstruct GridView: View {

        let id: Int //  检测数据是否对应

        let grid_row_num: Int
        let number: Int
        @Binding var nextNumber: Int
        @Binding var isPass: Bool
        @Binding var size: Int
        @Binding var time: Int
        @Binding var isStart: Bool
        @Binding var numberArray: [Int]
        var body: some View {
```

```
        Button(
            action: {
                if self.number == self.nextNumber {
                    if self.number == (self.grid_row_num * self.grid_row_num) {
                        self.isPass = true
                    } else {
                        self.nextNumber += 1
                    }
                }
            }
        ) {
            RoundedRectangle(cornerRadius: 9)
                .stroke(lineWidth: 3)
                .foregroundColor(.black)
                .frame(width: UIScreen.main.bounds.width / CGFloat(self.grid_row_num + 3), height:
UIScreen.main.bounds.width / CGFloat(self.grid_row_num + 3))
                .overlay{
                    Text("\(self.number)")
                        .font(.title3)
                        .foregroundColor(.black)
                }
        }
        .sheet(isPresented: $isPass, content: {
            PassView(size: $size, numberArray: $numberArray, nextNumber: $nextNumber, time: $time,
isPass: $isPass, isStart: $isStart)
                .highPriorityGesture(DragGesture())
        })
    }
}
//struct
//struct GridView_Previews: PreviewProvider {
//     static var previews: some View {
//         GridView(id: 1, grid_row_num: 10, number: 8)
//     }
//}//  TableView.swift
//  Shult
//  Created by xu on 2022/2/28.
//import SwiftUIstruct TableView: View {
//     @State var size = 5
//     @State var numberArray: [Int]
    @Binding var size: Int
    @Binding var numberArray: [Int]
    @Binding var nextNumber: Int

    @Binding var time: Int                    // 时间重头开始计时
    @Binding var isStart: Bool          // 返回

    @Binding var isPass: Bool
    var body: some View {
        VStack {
            ForEach(1...size, id:\.self) {i in
                HStack{
                    ForEach(1...size, id:\.self) {j in
//                          GridView(id:(i-1)*size+j-1, grid_row_num: size, number: numberArray[(i-
1)*size+j-1], nextNumber: $nextNumber)
                        GridView(id:(i-1)*size+j-1, grid_row_num: size, number: numberArray[(i-
1)*size+j-1], nextNumber: $nextNumber, isPass: $isPass, size: $size, time: $time, isStart: $isStart,
numberArray: $numberArray)
                    }
                }
            }
        }
    }
    func getRowCount(numberArray: [Int]) -> Int{
        return numberArray.count
    }
}//struct TableView_Previews: PreviewProvider {
//    static var previews: some View {
//        TableView()
//    }
//}
//  TimerView.swift
//  Shult
//  Created by xu on 2022/3/2.
//import SwiftUIstruct TimerView: View {
    @Binding var time: Int
    let timer = Timer.publish(every: 1, on: .main, in: .common).autoconnect()
    @Binding var isPass: Bool     var body: some View {
        VStack{
            HStack{
                Text("\(time)")
                    .font(.largeTitle)
                Text("s")
                    .font(.title2)
                    .offset(x: -5, y: 3)
            }
        }.onReceive(timer) { _ in
            if !isPass {
                self.time += 1
            }
//            if self.timeRemaining > 0 {
//                self.timeRemaining += 1
//            }
        }
    }
}
//struct ContentView: View {
//}//struct TimerView_Previews: PreviewProvider {
//    static var previews: some View {
//        TimerView()
//    }
//}
//  ButtonView.swift
//  Shult
//  Created by xu on 2022/3/2.
//import SwiftUI
//import UIKit.UITabVie
//struct ExampleView: View {
//    @State var text: String = ""
//    var body: some View {
//        UIKitTabView([
//            UIKitTabView.Tab(
//                view: NavView(),
//                barItem: UITabBarItem(title: "First", image: nil, selectedImage: nil)
//            ),
//            UIKitTabView.Tab(
//                view: Text("Second View"),
//                barItem: UITabBarItem(title: "Second", image: nil, selectedImage: nil)
//            )
//        ])
//    }
//}
//struct NavView: View {
//    var body: some View {
//        NavigationView {
//            VStack {
//                NavigationLink(destination: Text("This page stays when you switch back and forth between tabs (as expected on iOS)")) {
//                    Text("Go to detail")
//                }
//            }
//        }
//    }
```

```
//}//struct ButtonView: View {
//      let buttonName: String
//      var body: some View {
//          Button(
//              action: {
//                  print("test")
//              }
//          ) {
//              Capsule()
//                  .stroke(lineWidth: 3)
//                  .frame(width: 120, height: 40)
//                  .overlay{
//                      Text("\(buttonName)")
//                          .font(.title)
//                  }
//          }
//          .foregroundColor(.black)
//      }
//}//struct ButtonView_Previews: PreviewProvider {
//      static var previews: some View {
////            ButtonView(buttonName: "Start")

//          ExampleView（ ）

//      }
//}//   TypeTableView.swift
//   Shult
//   Created by xu on 2022/3/2.

//   宫格类型选择，如 4*4， 5*5import SwiftUIlet width = UIScreen.main.bounds.widthstruct

TypeTableView: View {

    @Binding var size: Int     // 每行显示数目，数据流向 TableView

    @Binding var numberArray: [Int] // 生成宫格，数据流向 TableView


    @State var state = -1

    var body: some View {
        VStack {


            Text("选择挑战等级")

                .font(.title)
                .padding()
                .overlay{
                    Divider()
                        .offset(x: 0, y: 35)
                }
//                .foregroundColor(.white)

//              HStack{
//                  TypeButton(id: 3, state: $state)
//                  TypeButton(id: 4, state: $state)
//              }
//              HStack{
//                  TypeButton(id: 5, state: $state)
//                  TypeButton(id: 6, state: $state)
//              }
//              HStack{
//                  TypeButton(id: 7, state: $state)
//                  TypeButton(id: 8, state: $state)
//              }
            HStack{
                TypeButton(id: 3, state: $state, size: $size, numberArray: $numberArray)
                TypeButton(id: 4, state: $state, size: $size, numberArray: $numberArray)
            }            HStack{
                TypeButton(id: 5, state: $state, size: $size, numberArray: $numberArray)
                TypeButton(id: 6, state: $state, size: $size, numberArray: $numberArray)
            }            HStack{
                TypeButton(id: 7, state: $state, size: $size, numberArray: $numberArray)
                TypeButton(id: 8, state: $state, size: $size, numberArray: $numberArray)
            }
        }
        .overlay{
            RoundedRectangle(cornerRadius: 9)
                .stroke(lineWidth: 3)
        }
    }
}struct TypeButton: View {
    var id: Int             // 判断变色并传递 size

    @Binding var state: Int   // 变色传递

    @Binding var size: Int    // 每行显示数目，数据流向 TableView

    @Binding var numberArray: [Int] // 生成宫格，数据流向 TableView
    var body: some View {
        Button(
            action: {
                self.state = self.id
                self.size = self.id
                self.numberArray = generateArray(n: self.id)
            }
        ) {
            Capsule()
                .stroke(lineWidth: 3)
                .frame(width: 120, height: 40)
                .overlay{
                    Text("\(self.id) × \(self.id)")
                }

        }
        .foregroundColor(self.id == self.state ? .black : .gray)
        .padding()
    }
}
//  Created by xu on 2022/3/3.
//import SwiftUIstruct GameView: View {

    @Binding var time: Int   //  计时器  对应  TimerView  视图

    @Binding var size: Int     //  每行显示数目，数据流向  TableView

    @Binding var numberArray: [Int] //  生成宫格，数据流向  TableView

    @Binding var isPass: Bool       @State var nextNumber = 1     //  下个数，数据流向  TableView ->

GridView
    @Binding var isStart: Bool
    var body: some View {
//          Text("hello")
        VStack{

            NextNumberView(nextNumber: $nextNumber)        //  下一个数                    Spacer()

            TimerView(time: $time, isPass: $isPass)
            Spacer()
//              TableView(size: $size, numberArray: $numberArray, nextNumber: $nextNumber, )
            TableView(size: $size, numberArray: $numberArray, nextNumber: $nextNumber, time:
$time, isStart: $isStart, isPass: $isPass)

            Spacer()
            Group{
                Button(
                    action: {
                        self.nextNumber = 1
                        self.numberArray = generateArray(n: self.size)
                        self.time = 0
                    }
                ) {
                    Capsule()
                        .stroke(lineWidth: 3)
                        .frame(width: 120, height: 40)
                        .overlay{
                            Text("重来")
                                .font(.title)
                        }
                }
                .foregroundColor(.black)


                Button(
                    action: {
                        self.nextNumber = 1
                        self.numberArray = generateArray(n: self.size)
                        self.time = 0
                        self.isStart = true
                    }
                ) {
                    Capsule()
                        .stroke(lineWidth: 3)
                        .frame(width: 120, height: 40)
                        .overlay{
                            Text("返回")
                                .font(.title)
                        }
                }
                .foregroundColor(.black)
            }

//            ButtonView(buttonName: "Again")
//            ButtonView(buttonName: "Return")
            Spacer()
        }//        TableView(size: 5)
```

```
    }
}
//struct GameView_Previews: PreviewProvider {
//      static var previews: some View {
//              GameView()
//      }
//}//    PassView.swift
//    Shult
//    Created by xu on 2022/3/3.
//import SwiftUIstruct PassView: View {

    @Binding var size: Int      // 保持不变

    @Binding var numberArray: [Int] // 生成同 size 大小的表格

    @Binding var nextNumber: Int       // 继续则下一个数字为 1

    @Binding var time: Int                  // 时间重头开始计时

    @Binding var isPass: Bool              // 记录是否弹窗

    @Binding var isStart: Bool           // 返回


    var passTime: Int {
        get {
            self.time
        }
    }
    var body: some View {
        VStack{
            HStack{
                Text("\(self.passTime)")
                    .font(.largeTitle)
                Text("s")
                    .font(.title2)
                    .offset(x: -5, y: 3)
            }
            .padding()


            HStack{
                Button(
                    action: {
                        self.isPass = false
                        self.numberArray = generateArray(n: self.size)
                        self.nextNumber = 1
                        self.time = 0
                    }
                ) {
                    Capsule()
                        .stroke(lineWidth: 3)
                        .frame(width: 120, height: 40)
                        .overlay{
                            Text("继续")
                                .font(.title)
                        }
                }
                .foregroundColor(.black)
                Button(
                    action: {
                        self.isStart = true
                        self.isPass = false
//                       self.numberArray = generateArray(n: self.size)
//                       self.time = 0
                    }
                ) {
                    Capsule()
                        .stroke(lineWidth: 3)
                        .frame(width: 120, height: 40)
                        .overlay{
                            Text("返回")
                                .font(.title)
                        }
                }
                .foregroundColor(.black)
                .padding()
            }
            .padding()
        }
        .padding()
        .overlay{
            RoundedRectangle(cornerRadius: 10)
                .stroke(lineWidth: 3)
                .padding()
        }
    }
```

```
}
struct PassView_Previews: PreviewProvider {
        static var previews: some View {
                PassView()
        }
}
```