Prelab 2

a) CreateDir.sh

```
#!/bin/bash
mkdir $1  # using 0 uses the script name
```

b) ShowDate.sh

```
#!/bin/bash
echo -n "The current date is $(date)"
```

c)
```
class WiimoteBtns {
private:
    int fd;
public:
    WiimoteBtns() {
        // Open Wiimote event file
        fd = open("/dev/input/event2", O_RDONLY);
        if (fd == -1) {
            std::cerr << "Error: Could not open event file - forgot sudo?\n";
            exit(1);
        }
    }

    // Close Wiimote event file
    ~WiimoteBtns() {
        close(fd);
        return 0;
    }

    void Listen() {
        while (true) {
            // Read a packet of 32 bytes from Wiimote
            char buffer[32];
            read(fd, buffer, 32);

            // Extract code (byte 10) and value (byte 12) from packet
            ButtonEvent(buffer[10], buffer[12]);
        }
    }

    void ButtonEvent(int code, int value) {
        // Print them
        std::cout << "Code = " << code << ", value = " << value << '\n';
    }
};
```

d)
```
class WiimoteAcell {
private:
    int fd; // Wiimote event file
public:
    WiimoteAcell() {
        // Open Wiimote event file
        fd = open("/dev/input/event0", O_RDONLY);
        if (fd == -1)
        {
            std::cerr << "Error: Could not open event file - forgot sudo?\n";
            exit(1);
        }
    }

    ~WiimoteAcell() {
        // Close Wiimote event file
        close(fd);
        return 0;
```

```
    }

    void Listen() {
        // Read a packet of 16 bytes from Wiimote
        char buffer[16];
        read(fd, buffer, 16);

        // Extract code (byte 10) and value (byte 12) from packet
        int code = buffer[10];
        short acceleration = * (short *) (buffer + 12);

        AccelerationEvent(code, acceleration);
    }

    virtual void AccelerationEvent(int code, short acceleration) {
        // Print them
        std::cout << "Code = " << code << ", acceleration = " << acceleration
<< '\n';
    }
};
```