Задание 1:

```cpp
class Attend {
    public:
        string IntMonthToString (int month) {
            switch (month) {
                case 1: {
                    return "jan";
                    break;
                }
                :
                case 12: {
                    return "dec")
                    break;
                }
                default: {
                    return "NaN",
                    break;
                }
            }
        };
        Attend & operator = (const Attend & right) {
            if (this == & right)
                return *(this)
            this -> fio = right.fio;
            this -> absent = right.absent;
            return *(this); };
```

```cpp
int & operator [] (const int month){
    string sMonth = IntMonthToString
    (month);
    if (absend.find(sMonth) != absend.
    end()){ return absend.at (sMonth
    }
};

void absend. addToAdsend (int month){
    string sMonth = IntMonthToStr (month)
    if (absend.find (sMonth) != absend.end()
    {
        absend.at (sMonth) += 1;
    } else {
        if (month < 13 && month > 0){
            absend.insert (pair<string, int>
            (sMonth, 1);
        }
    }
};
```

```
Задание 3;    bool Odd = true; mutex mut;
void printOdd(){
    int o = 0;
    while (o < 20){
        if (Odd){
            mut.lock();
            cout << o << endl;
            o += 2;
            Odd = false;
            mut.unlock();
        }
    }
}
void printOth(){
    int o = 0;
    while (o < 20){
        if (!Odd){
            mut.lock
            cout << o << endl;
            o += 2;
            Odd = true;
            mut.unlock() } } }
```

```cpp
int main (){
    thread thread1 (printOdd);
    thread thread2 (printOthr);
    thread1.detach();
    thread2.detach();
    return 0;
}
```