

主题一

一等奖

码启江城·畅行无界

```
if (openApp("i武汉")){  
    // 指尖一码，江城无界  
    const auto& traffic=getRealTimeTraffic();  
    // 数织三镇路  
    cityPass oneCode=integrateAllModes  
    (traffic, bus , metro, ferry, tram);  
    //一码融百端，出行不折返  
    myLife=TransitService::easyOutgoing(oneCode).withNoHurry();  
    //智联时光，从容如常
```

二等奖

看见

```
def robot_vision(see_you):  
    if see_you:  
        return "See the world";  
    else:  
        return "See myself"
```

心跳协议

```
if(network.ping(heartbeat)){ //当网络轻触心跳  
    devices.sync(empathy); //万物同步共情  
    world.compile(love);} //世界以爱编译
```

三等奖

云端守望

```
cloud.watch(EmergencyHub) while latency < 20ms
doctors.call(new RemoteClinic()).resolve(LIFE_LINE)
Legacy(TECH_HEAT).launch(T ELEMEDICINE)
//毫秒级响应，让急救不被距离阻隔
//远程诊室里，代码托举生命重量
//技术有温度，让健康触手可及
```

萤火归途 数字守护

```
sensors = SenseCity.pulse()
# 十亿级感知节点实时共鸣
if find_elderly_stray():
# 当感知到长者徘徊时
street_light.slowly_guide_home()
# 街灯如萤火虫连成归途
```

字节心脉

```
infrastructure=lambda nodes:sum(nodes)
#象征 5G/6G 构建的全域互联网络
warmth=lambda value:float('inf')if value==0 else value
#人文让空洞的技术（0）拥有无限价值
symbiosis=lambda tech,human:infrastructure([tech,human])+warmth(human)
#骨与魂相融，才是有温度的万物智联
```

主题二 一等奖

数字流沙·暖熵减

```
const moments = new Map();
watchLife((moment, emotion) => {
  if(emotion.score() > 0.8) {
    moments.set(Date.now(), moment.compress('warmth'));
    console.log(`已存档: ${moment}, 温度: ${emotion.temperature}°C`);
  }
});
// 时光如流沙, 代码作容器
// 2025的暖意存入哈希表的每一格
// 待2026的清晨, 解压成第一缕阳光
```

二等奖

幸福的增量迭代

```
for fragment in memories_2025.query("tag == 'warmth'"):
  reward = reward = fragment_exture("happiness_iaure")
Year_2026.model.update_weights(learning_rate = reward)
Year_2026.run_forever(mode = "optimistic")
# 溯源 2025 年账簿, 遍历每一笔带有“暖意”标签的原始凭证
# 提取幸福特征值, 将其作为驱动未来的核心学习率
# 让旧岁里的平凡碎屑, 在万物智联的算法里实现指数级增值
```

指尖拾岁暖, 码上寄晨光

```
float*memory=new float[365];
while(day++<365){
future+=(*memory++=collect_warmth()); const
float*hope=memory-1;}
//拾忆为匣, 盛纳一岁温柔
//似水流年, 撷取朝夕美好
//暖意入格, 步履向前; 温情融川, 奔赴来日
//记忆回溯, 定格此瞬; 希冀不改, 岁岁长安
```

三等奖

把微光装进心形匣，寄给2026

```
pearls = ['早点的香', '电车准点', '窗边雨声', '留言未读']
chest = {'2025': [p + '♡' for p in pearls]}
print('致2026: ' + ','.join(s.replace('未读', '已读') for s in
    chest['2025']) + ', 愿它们继续生长')
# 拾光成珠：把一年微小的温柔逐一命名
# 心形加密：为每粒微光刻下热度与回响
# 温柔兑现：把未读的思念读完
```

时光存档：智能暖日常

```
string archive[] = {"smart_window_light", "smart_speaker_laugh",
    "video_call_reunion"};
for(int i=0; i<3; i++) cout << "2025: " << archive[i] << "\n";
cout << "2026: All warm memories load \n";
// 存档2025的温暖瞬间
// 加载至2026的序章里
// 让美好循环，温柔不休
```

晨鸣唤新程，暖意赴朝夕

```
if (alarm.triggered() && weather.isSunny()) {
    speaker.play("Good morning, 2026!");
    window.curtain.openGradually();}
// 当智能闹钟触发，且室外为晴天时
// 智能音箱播放温馨问候
// 智能窗帘缓缓拉开
// 愿每一次苏醒，都被世界温柔以待 —— 期许新的一年，每一次醒来都能被生活的美好与温柔包裹
```

褶皱的抚平者

```
day_canvas = blend(schedule, my_rhythm, campus_weather)
time_fold = detect_conflict(day_canvas, key='noon', empathy=True)
healing_plan = prescribe(time_fold, with='pre_order', and='by_the_lake')
```

将课表的锋利、我的困倦与梧桐叶隙的阳光，糅合成今日的画布

敏锐地找到画布上那处紧张的褶皱——它正勒疼一个急需喘息的午后

于是开出处方：一份提前抵达的餐食，与一片可供呼吸的湖