实验五

求子代最优个体和平均适应度

代码：

#include<stdio.h>  
#include<time.h>  
#include<stdlib.h>  
typedef struct  
{  
int code;  //染色体  
int degree;//适应度  
}Indi;  
Indi group[40];//种群规模为40  
void Judge(Indi &x)  
{  
x.degree=x.code\*x.code;  
}  
int happened(double p)//发生一个p=0~1间概率的事件  
{  
return rand()<(int)(p\*RAND\_MAX);  
}  
void Cross(Indi &x,Indi &y)//交叉操作  
{  
Indi z,z1;  
int temp,temp1;  
temp=x.code&0x3;  
temp1=y.code&0x3;  
z.code=x.code-temp+temp1;  
z1.code=y.code-temp1+temp;  
Judge(z);  
Judge(z1);  
if(x.degree<y.degree)  
{  
if(z.degree>=x.degree) //如果新个体不如双亲，淘汰之  
x=z;  
}  
else  
{  
if(z.degree>=y.degree)  
y=z;  
}  
if(x.degree<y.degree)  
{  
if(z1.degree>=x.degree) //如果新个体不如双亲，淘汰之  
x=z1;  
}  
else  
{  
if(z1.degree>=y.degree)  
y=z1;  
}  
}  
int main(void)  
{  
Indi indidest;  
int i,j,best,x,y,c;  
int sum,strick,SUM=0;  
static int n=0;  
srand(time(NULL));  
for(i=0;i<40;++i)//随机得到初始种群  
{  
group[i].code=rand()%32;  
Judge(group[i]);  
}  
for(i=1;i<=10;++i)//固定进化10代  
{  
for(sum=0,best=0,j=0;j<40;++j)  
{  
sum+=group[j].degree;//求总的适应度sum  
if(group[j].degree>group[best].degree)  
{  
best=j;//求当前最优个体  
}  
}  
printf("第%2d代中 最优个体为 %d  (%d) 平均适应度为 %10f\n",  
i,group[best].code,group[best].degree,sum/40.0);  
for(c=40;c;--c)  
{  
strick=(int)((float)rand()/RAND\_MAX\*sum);  //赌盘中的色子，选择个体x,y  
for(x=0;x<40&&strick>=group[x].degree;++x)  
strick-=group[x].degree;  
strick=(int)((float)rand()/RAND\_MAX\*sum);  
for(y=0;y<40&&strick>=group[y].degree;++y)  
strick-=group[y].degree;  
if(happened(0.9))  
Cross(group[x],group[y]);//交叉  
}  
}  
return 0;  
}

运行结果：

