

به نام خدا

• داکيومنت مربوط به توضیحات پروژه گسسته:

• ارائه دهندگان : علی دشت بزرگ - علیرضا سعیدنیا

توضیحات :

کلاس Geraph :

فیلدها :

`Boolean[][]list`----> ماتریس مجاورت برای گرافها

`Int lines`---> تعداد خطوط هر گراف

کانستراکتورها:

`Geraph(Boolean[]list,int lines)`

`Geraph(int points,int lines)`

متدها:

`void setList (Scanner scanner)`---> با توجه به داده های کاربر لیست مجاورت را تنظیم میکند

`Boolean [][] getList()`

void setList(Boolean[][]list)

int getLines()

void setLines(int lines)

کلاس Engine: اینترفیس Runnable را پیاده سازی میکند

Geraph geraph: گراف مسئله

Integer min: مقدار محاسبه شده برای کوتاهترین دور

Integer Point: نقطه ای که برای آن کوتاهترین دور را پیدا میکنیم

Boolean Stop: یک نشانگر برای توقف عملیات محاسبه

HashMap<Boolean[],Integer>skip: یک لیست برای کمک در محاسبه کوتاهترین مسیر که هر

راس را با مجاورتهایش ذخیره میکند

کانستراکتورها:

Engine (Geraph geraph,Integer point)

متود ها:

Void setSkip(): با توجه به گراف لیست راس و مجاورتهایش را میسازد

Geraph getGeraph()

void setGeraph(Geraph geraph)

Integer getPoint()

Integer getMin()

void setMin(Integer min)

Void goThrough(int skip,int point , Boolean[]lines)

Skip : راسی که skip میشود ، اگر این نباشد فقط روی یک یال عقب جلو میکند

Point : راس هدف

Lines : لیست مجاورت برای راسی که هستیم

کار اصلی این تابع حرکت کردن از یک راس به راس دیگر با استفاده از Recursive است تا اینکه بتواند به راس ابتدایی یا همان راس هدف برگردد

Void Run() : این متد برای استفاده از multi thread پیاده سازی میشود و اولین فراخوانی متد goThrough() در اینجا است که از راس اول به راس دوم حرکت میکند بعد از محاسبه کمترین مسیر در min ذخیره میشود ، البته اگر کمترین مسیر پیدا نشود مقدار ۹۹۹۹ ذخیره میشود

کلاس Main :

متد Main : گراف ساخته میشود و ArrayList از Engine برای محاسبه کوتاهترین دور برای هر راس ساخته میشود و سپس بعد از انجام محاسبه MultiThread کمترین مقدار محاسبه شده نمایش داده میشود اگر کمترین موجود نباشد عبارت no orbit found چاپ میشود

تست برنامه :

```
6 5
1 2
2 3
3 1
4 5
3 5
3

Process finished with exit code 0
```

```
6 3
1 2
1 6
1 4
No orbit found

Process finished with exit code 0
```

```
4 4
1 2
2 3
3 4
4 1
4
Process finished with exit code 0
```

```
5 7
1 2
1 3
1 4
1 5
2 3
2 4
2 5
3
Process finished with exit code 0
```

```
4 1
1 2
No orbit found

Process finished with exit code 0
```