

# Мухамадиев Владимир

## Задание 2

### Загрузка и предварительная обработка

```
In[1]:= (*Get["https://raw.githubusercontent.com/szhorvat/IGraphM/master/IGInstaller.m"] *)  
      |  
      |_взять  
  
In[2]:= Needs["IGraphM`"]  
      |  
      |_необходимо  
  
IGraph/M 0.3.114 (February 14, 2020)  
Out[2]:= Evaluate IGDokumentation[] to get started.  
  
In[3]:= data1 = Import[NotebookDirectory[] <> "\\asoiaf-book1-edges.csv", "CSV"];  
      |  
      |_импорт |директория файла блокнота  
  
In[4]:= e11 = Table[data1[[i, 1]] ↔ data1[[i, 2]], {i, 2, Length[data1]}];  
      |  
      |_таблица значений |длина  
  
In[5]:= ew1 = data1[[2 ;; All, 4]];  
      |  
      |_всё  
  
In[6]:= g1 = Graph[e11, EdgeWeight → ew1];  
      |  
      |_граф |вес ребра  
  
In[7]:= data2 = Import[NotebookDirectory[] <> "asoiaf-book2-edges.csv", "CSV"];  
      |  
      |_импорт |директория файла блокнота  
  
In[8]:= e12 = Table[data2[[i, 1]] ↔ data2[[i, 2]], {i, 2, Length[data2]}];  
      |  
      |_таблица значений |длина  
  
In[9]:= ew2 = data2[[2 ;; All, 4]];  
      |  
      |_всё  
  
In[10]:= g2 = Graph[e12, EdgeWeight → ew2];  
      |  
      |_граф |вес ребра  
  
In[11]:= data3 = Import[NotebookDirectory[] <> "asoiaf-book3-edges.csv", "CSV"];  
      |  
      |_импорт |директория файла блокнота  
  
In[12]:= e13 = Table[data3[[i, 1]] ↔ data3[[i, 2]], {i, 2, Length[data3]}];  
      |  
      |_таблица значений |длина  
  
In[13]:= ew3 = data3[[2 ;; All, 4]];  
      |  
      |_всё  
  
In[14]:= g3 = Graph[e13, EdgeWeight → ew3];  
      |  
      |_граф |вес ребра  
  
In[15]:= data4 = Import[NotebookDirectory[] <> "asoiaf-book4-edges.csv", "CSV"];  
      |  
      |_импорт |директория файла блокнота  
  
In[16]:= e14 = Table[data4[[i, 1]] ↔ data4[[i, 2]], {i, 2, Length[data4]}];  
      |  
      |_таблица значений |длина  
  
In[17]:= ew4 = data4[[2 ;; All, 4]];  
      |  
      |_всё
```

```

In[18]:= g4 = Graph[e14, EdgeWeight → ew4];
           |граф      |вес ребра

In[19]:= data5 = Import[NotebookDirectory[] <> "asoiaf-book5-edges.csv", "CSV"];
           |импорт   |директория файла блокнота

In[20]:= e15 = Table[data5[[i, 1]] ↔ data5[[i, 2]], {i, 2, Length[data5]}];
           |таблица значений      |длина

In[21]:= ew5 = data5[[2 ;; All, 4]];
           |всё

In[22]:= g5 = Graph[e15, EdgeWeight → ew5];
           |граф      |вес ребра

In[23]:= gfullprep = Normal[
           |нормальное выражение
           GroupBy[{Flatten[{e11, e12, e13, e14, e15}], Flatten[{ew1, ew2, ew3, ew4, ew5}]}],
           |сгруппир... |уплостить
           First → Last, Total];
           |первый |пос... |суммировать

In[24]:= elfull = gfullprep[[All, 1]];
           |всё

In[25]:= ewfull = gfullprep[[All, 2]];
           |всё

In[26]:= gfull = Graph[elfull, EdgeWeight → ewfull];
           |граф      |вес ребра

In[27]:= RankedByCentrality[graph_, centralityfunction_] := Transpose[Prepend[
           |транспози... |добавить в начало
           Transpose[ReverseSortBy[{VertexList[graph], centralityfunction[graph]}], Last]],
           |транспози... |сортировка в обр... |список вершин графа
           Range[VertexCount[graph]]];
           |диап... |число вершин

In[28]:= SetAttributes[RankedByCentrality, Listable]
           |задаты атрибуты      |автоматически

```

## 1. Определите топ-10 персонажей по значению центральности по степени. Сколько среди них Старков?

```

In[29]:= topD = RankedByCentrality[{g1, g2, g3, g4, g5}, DegreeCentrality];
           |центральность по степени

In[30]:= topDfull = RankedByCentrality[gfull, DegreeCentrality];
           |центральность по степени

In[31]:= topD[[1, 1 ;; 10]]
Out[31]= {{1, Eddard-Stark, 66}, {2, Robert-Baratheon, 50}, {3, Tyrion-Lannister, 46},
           {4, Catelyn-Stark, 43}, {5, Jon-Snow, 37}, {6, Sansa-Stark, 35}, {7, Robb-Stark, 35},
           {8, Bran-Stark, 32}, {9, Joffrey-Baratheon, 30}, {10, Cersei-Lannister, 30}}

In[32]:= Length[Cases[topD[[1, 1 ;; 10]], {_, x_String, _} /; StringContainsQ[x, "-Stark"]]]
           |длина |случаи по образцу      |содержится в строке?
Out[32]= 5

```

## 2. Какое значение взвешенной степени у Eddard-Stark в сети?

```
In[33]:= WeightedDegreeCentrality[graph_] := Total[Normal[WeightedAdjacencyMatrix[graph]]]
           |сумма...|норма...|матрица весов графа

In[34]:= topWD = RankedByCentrality[{g1, g2, g3, g4, g5}, WeightedDegreeCentrality];

In[35]:= topWDFull = RankedByCentrality[gfull, WeightedDegreeCentrality];

In[36]:= FirstCase[topWD[[1]], {_, "Eddard-Stark", _}] [[3]]
           |первый по образцу

Out[36]= 1284
```

## 3. Сколько персонажей из топ-10, определенным по центральности по степени (из вопроса 1) осталось в топ-10 по взвешенной степени ?

```
In[37]:= Length[topD[[1, 1 ;; 10, 2]] ∩ topWD[[1, 1 ;; 10, 2]]
           |длина

Out[37]= 8
```

## 4. Какой новый персонаж появился в топ-10 персонажей по значению центральности по посредничеству?

```
In[38]:= topB = RankedByCentrality[{g1, g2, g3, g4, g5}, BetweennessCentrality];
           |центральность по посредничест

In[39]:= topBfull = RankedByCentrality[gfull, BetweennessCentrality];
           |центральность по посредничест

In[40]:= Complement[topB[[1, 1 ;; 10, 2]],
           |дополнение
           topD[[1, 1 ;; 10, 2]] ∪ topWD[[1, 1 ;; 10, 2]] ∩ topB[[1, 1 ;; 10, 2]]

Out[40]= {Drogo}
```

## 5. Постройте топ-10 персонажей по значению центральности по посредничеству с учетом веса ребер. Кто теперь возглавляет список?

```
In[41]:= topBW = RankedByCentrality[{g1, g2, g3, g4, g5}, IGBetweenness];

In[42]:= topBW[[1, 1, 2]]

Out[42]= Robert-Baratheon
```

## 6. Постройте топ-10 по значению PageRank. Какое место в топе занимает Daenerys-Targaryen?

```
In[43]:= topPRW = RankedByCentrality[{g1, g2, g3, g4, g5}, IGPagerank];
In[44]:= topPRWfull = RankedByCentrality[gfull, IGPagerank];
In[45]:= topPRW[[1, 1 ;; 10]]
Out[45]= {{1, Eddard-Stark, 0.072394}, {2, Robert-Baratheon, 0.0485173},
          {3, Jon-Snow, 0.0477069}, {4, Tyrion-Lannister, 0.0436744},
          {5, Catelyn-Stark, 0.034667}, {6, Bran-Stark, 0.0297742},
          {7, Robb-Stark, 0.0292162}, {8, Daenerys-Targaryen, 0.0270896},
          {9, Sansa-Stark, 0.0269618}, {10, Cersei-Lannister, 0.0216317}}
```

```
In[46]:= FirstCase[topPRW[[1]], {_, "Daenerys-Targaryen", _}][[1]]
          первый по образцу
Out[46]= 8
```

## 7. Постройте рейтинг по значению центральности по степени для персонажей 5-ой книги. Какую теперь наивысшую строчку рейтинга занимает персонаж из Дома Старков?

```
In[47]:= topD[[5, 1 ;; 10]]
Out[47]= {{1, Jon-Snow, 62}, {2, Daenerys-Targaryen, 58},
          {3, Stannis-Baratheon, 47}, {4, Tyrion-Lannister, 33},
          {5, Theon-Greyjoy, 33}, {6, Cersei-Lannister, 28}, {7, Barristan-Selmy, 25},
          {8, Hizdahr-zo-Loraq, 22}, {9, Asha-Greyjoy, 18}, {10, Melisandre, 17}}
```

```
In[48]:= FirstCase[topD[[5]], {_, x_String, _} /; StringContainsQ[x, "-Stark"]][[1]]
          первый по образцу                               содержится в строке?
Out[48]= 21
```

## 8. Выберите персонажа и построьте график, показывающий как менялась его влияние от номера книги.

```
In[49]:= topEV = RankedByCentrality[{g1, g2, g3, g4, g5}, EigenvectorCentrality];
          центральность собственного век
In[50]:= topEVfull = RankedByCentrality[gfull, EigenvectorCentrality];
          центральность собственного век
In[51]:= topWC = RankedByCentrality[{g1, g2, g3, g4, g5}, ClosenessCentrality];
          центральность по близости
In[52]:= topWCfull = RankedByCentrality[gfull, ClosenessCentrality];
          центральность по близости
In[53]:= perIAB =
          VertexList[g1] ∩ VertexList[g2] ∩ VertexList[g3] ∩ VertexList[g4] ∩ VertexList[g5];
          список вершин графа список вершин графа список вершин графа список вершин графа список вершин графа
```

```
In[54]:= per = RandomChoice[perIAB]
```

случайный выбор

```
Out[54]= Jeor-Mormont
```

```
In[55]:= RankedByName[top_, name_] :=
  {Range[Length[top]], FirstCase[top[[#]], {_, name, _}][[1]] & /@ Range[Length[top]]}^T
```

диап... длина

первый по образцу

диап... длина

```
In[56]:= Module[{data = RankedByName[#, per] & /@ {topD, topWD, topEV, topPRW, topWC, topB}},
  программный модуль
```

```
ListLinePlot[data, ScalingFunctions -> "Reverse",
```

линейный график дан... масштабирующие функ... расположить в обратном порядке

```
AxesOrigin -> {1, Max[Flatten[data, 1][[All, 2]]]},
```

точка пересечения... ма... упростить всё

```
PlotRange -> {All, {1, Max[Flatten[data, 1][[All, 2]]]}, AxesLabel -> {"Book", "Rank"},
```

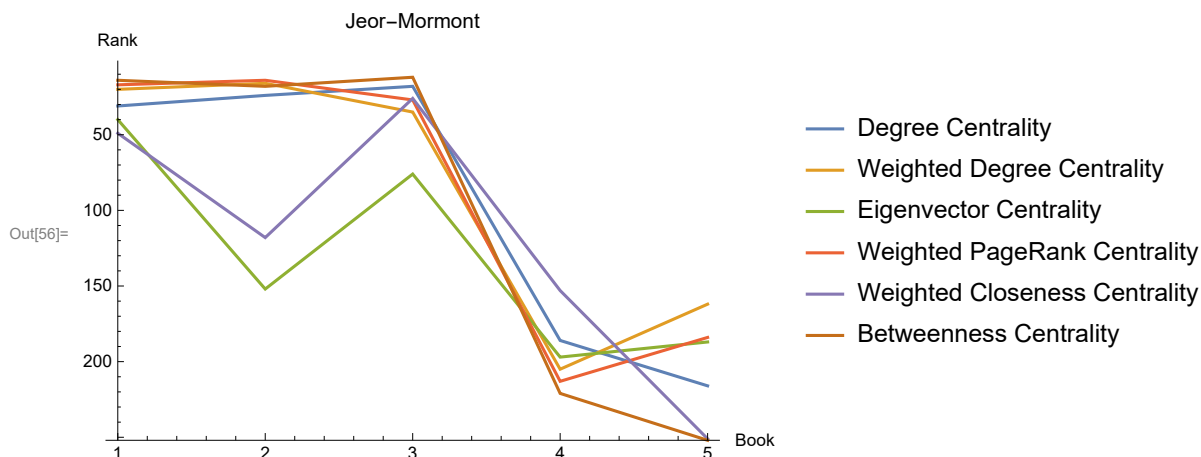
отображаемы... всё ма... упростить всё обозначения на осях

```
PlotLabel -> per, PlotLegends -> {"Degree Centrality", "Weighted Degree Centrality",
```

пометка графика легенды графика градус градус

```
"Eigenvector Centrality", "Weighted PageRank Centrality",
```

```
"Weighted Closeness Centrality", "Betweenness Centrality"]}]
```



## 9. Загрузите таблицу наиболее влиятельных персонажей, определенных по объединенной сети.

```

In[57]:= Grid[Prepend[Prepend[Partition[Flatten[SortBy[
  Table[{VertexList[gfull][i], (FirstCase[#, {_, VertexList[gfull][i], _}] & /@
    {topDfull, topWDFull, topEVfull, topPRWfull, topWCfull, topBfull})]
  All, 1]], {i, 1, VertexCount[gfull]}], Total[#[[2]] &][[1 ;; 10]], 7],
  {SpanFromAbove, "Degree", "Weighted Degree", "Eigenvector",
    "Weighted PageRank",
    "Weighted Closeness", "Betweenness"}],
  {"Name", "Ranked By Centrality", SpanFromLeft}],
  Background → {None, {{{Pink, Lighter[Blue, 0.7]}}, {1 → Gray, 2 → Gray}}},
  Dividers →
  All, Spacings →
  {1, 1}]

```

Name	Ranked By Centrality					
	Degree	Weighted Degree	Eigenvector	Weighted PageRank	Weighted Closeness	Betweenness
Tyrion-Lannister	1	1	1	2	7	2
Jaime-Lannister	3	7	3	5	1	6
Cersei-Lannister	4	3	2	3	8	7
Jon-Snow	2	2	14	1	10	1
Stannis-Baratheon	5	13	8	8	3	5
Eddard-Stark	10	5	7	6	13	9
Robert-Baratheon	14	10	6	13	2	10
Joffrey-Baratheon	12	4	4	9	11	20
Arya-Stark	6	11	11	7	20	8
Sansa-Stark	7	8	5	12	19	13

```

In[58]:= Grid[Prepend[
  таб...  [добавить в начало
    Transpose[Prepend[{topDfull[[1, 2 ;; 3]], topWDfull[[1, 2 ;; 3]], topEVfull[[1, 2 ;; 3]],
      транспози... [добавить в начало
        topPRWfull[[1, 2 ;; 3]], topWCfull[[1, 2 ;; 3]], topBfull[[1, 2 ;; 3]]}^T,
        {"Degree", "Weighted Degree", "Eigenvector", "Weighted PageRank",
          градус      градус
          "Weighted Closeness", "Betweenness"}]], {"Centrality", "№1", "Value"}]],
  Background → {None, {{Pink, Lighter[Blue, 0.7]}}, {1 → Gray}}},
  фон      ни одног... роз... более с... синий      серый
  Dividers → All, Spacings → {1, 1}]
разделители всё размер зазора

```

Centrality	№1	Value
Degree	Tyrion-Lannister	122
Weighted Degree	Tyrion-Lannister	2873
Eigenvector	Tyrion-Lannister	0.0188473
Weighted PageRank	Jon-Snow	0.0357054
Weighted Closeness	Jaime-Lannister	0.0958755
Betweenness	Jon-Snow	60635.8

Out[58]=