Исходные данные:

int n; // Число команд

int nl; // Число команд лидеров

int[] s; // Номера зрелищных туров

int d; // Число дней

int f; // Минимальное число игр за один игровой день

int g; // Максимальное число игр за один игровой день

int[] q; // Номера дней недели

// Номера приоритетных дней недели с числом в 4 тура для команд

struct V { public int d, n; public V(int d, int n) { this.d = d; this.n = n; } }

V[][] v;

int[][] t; // Номера прилоритетных часов

int[,,] x; // Номера команд в матчах [2 \* (n - 1), n / 2, 2]

int[,] y; // Номера дней проведения матчей [2 \* (n - 1), n / 2]

int[,] z; // Номера часов проведения матчей [2 \* (n - 1), n / 2]

Алгоритмы решения:

// Составление расписания матчей по турам

int[,,] GetTours(int n, int nl, int[] s)

{

int[,,] x = new int[2 \* (n - 1), n / 2, 2];

// Туры

int[][,] a = new int[n - 1][,];

for (int i = 0; i < n - 1; i++)

a[i] = new int[n / 2, 2];

{

// Лента команд

int[] b = new int[n];

for (int i = 0; i < n; i++)

b[i] = i;

// Заполнение туров в неустановленном порядке

for (int i = 0; i < n - 1; i++)

{

for (int j = 0; j < n / 2; j++)

{

a[i][j, 0] = b[j];

a[i][j, 1] = b[n - 1 - j];

}

if (i < n - 2)

{

int temp = b[n - 1];

for (int k = n - 1; k > 1; k--)

b[k] = b[k - 1];

b[1] = temp;

}

}

}

{

// Количества игр между командами лидерами в турах

int[] b = new int[n - 1];

for (int i = 0; i < n - 1; i++)

for (int j = 0; j < n / 2; j++)

if ((a[i][j, 0] < nl) && (a[i][j, 1] < nl))

b[i]++;

// Сортировка туров по количеству игр между командами лидерами

Array.Sort(b, a);

Array.Reverse(a);

}

{

// Распределение зрелищных туров по кругам

List<int>[] b = { new List<int>(), new List<int>() };

for (int i = 0; i < s.Length; i++)

{

if (s[i] < n - 1)

b[0].Add(s[i]);

else

b[1].Add(s[i]);

}

// Заполнение расписания

for (int i = 0; i < 2; i++)

{

// Заполнение зрелищных туров в порядке сортировки

for (int j = 0; j < b[i].Count; j++)

{

for (int k = 0; k < n / 2; k++)

for (int l = 0; l < 2; l++)

x[b[i][j], k, l] = a[j][k, l];

}

// Заполнение оставшихся туров

int m = 0; // Номер свободного тура

for (int j = b[i].Count; j < n - 1; j++)

{

// Поиск свободного тура

for (; x[i \* (n - 1) + m, 0, 0] != x[i \* (n - 1) + m, 0, 1]; m++) ;

// Заполнение свободного тура

for (int k = 0; k < n / 2; k++)

for (int l = 0; l < 2; l++)

x[i \* (n - 1) + m, k, l] = a[j][k, l];

}

}

}

return x;

}

// Составление расписания матчей по дням

int[,] GetDays(int n, int d, int f, int g, int[] q, V[][] v, int[,,] x)

{

int[,] y = new int[2 \* (n - 1), n / 2];

// Присвоение расписанию неустановленных значений

for (int i = 0; i < 2 \* (n - 1); i++)

for (int j = 0; j < n / 2; j++)

y[i, j] = -1;

// Счетчик выполнения предпочтений

int[][][] c = new int[n][][];

for (int i = 0; i < n; i++)

{

c[i] = new int[v[i].Length][];

for (int k = 0; k < v[i].Length; k++)

{

c[i][k] = new int[2];

c[i][k][0] = v[i][k].d;

}

}

int[] o = new int[d]; // Счетчик матчей в день

int lastd = 0; // День последнего размещенного матча

for (int i = 0; i < 2 \* (n - 1); i++)

{

int a = lastd \* g + o[lastd]; // Первый доступный слот для текущего тура

int b = (d \* g - a) / (2 \* (n - 1) - i); // Число выделенных слотов для текущего тура

// Сброс счетчика выполнения предпочтений каждые 4 тура

if (i % 4 == 0)

{

for (int j = 0; j < n; j++)

for (int k = 0; k < v[j].Length; k++)

c[j][k][1] = v[j][k].n;

}

// Размещение по предпочтениям

for (int j = 0; j < n / 2; j++)

{

for (int k = 0; k < 2; k++)

{

for (int l = 0; l < c[x[i, j, k]].Length; l++)

{

if (c[x[i, j, k]][l][1] > 0)

{

for (int e = a / g; e <= (a + b - 1) / g; e++)

{

if ((q[e] == c[x[i, j, k]][l][0]) && (o[e] < g))

{

if (e == (a + b - 1) / g)

{

int ol = b % g;

if (ol == 0)

ol = g;

if (o[e] >= ol)

break;

}

y[i, j] = e;

if (e > lastd)

lastd = e;

o[e]++;

c[x[i, j, k]][l][1]--;

if (k == 0)

for (l = 0; l < c[x[i, j, 1]].Length; l++)

if ((q[e] == c[x[i, j, 1]][l][0]) && (c[x[i, j, 1]][l][1] > 0))

{

c[x[i, j, 1]][l][1]--;

break;

}

break;

}

}

if (y[i, j] >= 0)

break;

}

}

if (y[i, j] >= 0)

break;

}

}

// Размещение без предпочтений

{

int j;

int e = a / g;

// Размещение в игровые дни с недостающим числом игр

for (j = 0; j < n / 2; j++)

{

if (y[i, j] < 0)

{

for (; e < lastd; e++)

if ((o[e] > 0) && (o[e] < f))

{

y[i, j] = e;

o[e]++;

break;

}

if (e == lastd)

break;

}

}

// Размещение в остальные дни

for (; j < n / 2; j++)

{

if (y[i, j] < 0)

for (e = a / g; e <= (a + b - 1) / g; e++)

if (o[e] < g)

{

y[i, j] = e;

if (e > lastd)

lastd = e;

o[e]++;

break;

}

}

}

// Выполнение ограничения на минимальное число игр в игровой день

for (int j = a / g; j < lastd; j++)

{

while ((o[j] > 0) && (o[j] < f))

for (int k = 0; k < n / 2; k++)

if (y[i, k] == lastd)

{

y[i, k] = j;

o[lastd]--;

o[j]++;

if (o[lastd] == 0)

for (int l = lastd - 1; ; l--)

if (o[l] > 0)

{

lastd = l;

break;

}

break;

}

}

}

return y;

}

// Составление расписания матчей по часам

int[,] GetHours(int n, int d, int g, int[][] t, int[,,] x, int[,] y)

{

int[,] z = new int[2 \* (n - 1), n / 2];

// Присвоение расписанию неустановленных значений

for (int i = 0; i < 2 \* (n - 1); i++)

for (int j = 0; j < n / 2; j++)

z[i, j] = -1;

bool[,] a = new bool[d, g]; // Метки занятых часов

for (int i = 0; i < 2 \* (n - 1); i++)

{

// Первый день следующего тура

int b = d;

if (i < 2 \* (n - 1) - 1)

for (int j = 0; j < n / 2; j++)

if (y[i + 1, j] < b)

b = y[i + 1, j];

// Число игр следующего тура в первый день

int c = 0;

if (i < 2 \* (n - 1) - 1)

for (int j = 0; j < n / 2; j++)

if (y[i + 1, j] == b)

c++;

// Размещение по предпочтениям

for (int j = 0; j < n / 2; j++)

{

for (int k = 0; k < 2; k++)

{

for (int l = 0; l < t[x[i, j, k]].Length; l++)

{

if ((a[y[i, j], t[x[i, j, k]][l]] == false) && ((y[i, j] < b) || (t[x[i, j, k]][l] < g - c)))

{

z[i, j] = t[x[i, j, k]][l];

a[y[i, j], t[x[i, j, k]][l]] = true;

break;

}

}

if (z[i, j] >= 0)

break;

}

}

// Размещение без предпочтений

for (int j = 0; j < n / 2; j++)

{

if (z[i, j] == -1)

for (int k = 0; ; k++)

{

if (a[y[i, j], k] == false)

{

z[i, j] = k;

a[y[i, j], k] = true;

break;

}

}

}

}

return z;

}

**Пример решения**

Исходные параметры

n = 6;

nl = 3;

s = new int[] { 3, 4, 8 };

d = 16;

f = 2;

g = 3;

q = new int[] { 0, 1, 0, 0, 0, 4, 1, 0, 0, 2, 0, 4, 0, 3, 2, 0 };

t = new int[n][];

t[0] = new int[] { 1, 2 };

t[1] = new int[] { 1 };

t[2] = new int[] { 0 };

t[3]= new int[0];

t[4]= new int[0];

t[5]= new int[] { 1 };

v = new V[n][];

v[0] = new V[] { new V(1, 3) };

v[1] = new V[] { new V(1, 2), new V(4, 2) };

v[2] = new V[] { new V(3, 2)};

v[3] = new V[0];

v[4] = new V[0];

v[5] = new V[] { new V(1, 3) };

Составление расписания матчей по турам

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |
| 5 | 4 | 3 |

0 тур: (0,5), (1, 4), (2,3)

|  |  |  |
| --- | --- | --- |
| 0 | 5 | 1 |
| 4 | 3 | 2 |

1 тур: (0,4), (5, 3), (1,2)

…

Туры в неустановленном порядке:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 |
| 0,5 | 0,4 | 0,3 | 0,2 | 0,1 |
| 1,4 | 5,3 | 4,2 | 3,1 | 2,5 |
| 2,3 | 1,2 | 5,1 | 4,5 | 3,4 |

Количества игр между командами лидерами:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 |
| 0 | 1 | 0 | 1 | 1 |

Сортировка туров по количеству игр между командами лидерами:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 |
| 0,1 | 0,2 | 0,4 | 0,3 | 0,5 |
| 2,5 | 3,1 | 5,3 | 4,2 | 1,4 |
| 3,4 | 4,5 | 1,2 | 5,1 | 2,3 |

Заполнение зрелищных туров:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 |
|  |  |  | 0,1 | 0,2 |
|  |  |  | 2,5 | 3,1 |
|  |  |  | 3,4 | 4,5 |

Заполнение оставшихся туров:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 |
| 0,4 | 0,3 | 0,5 | 0,1 | 0,2 |
| 5,3 | 4,2 | 1,4 | 2,5 | 3,1 |
| 1,2 | 5,1 | 2,3 | 3,4 | 4,5 |

Аналогично для второго круга:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0,4 | 0,3 | 0,5 | 0,1 | 0,2 | 0,2 | 0,4 | 0,3 | 0,1 | 0,5 |
| 5,3 | 4,2 | 1,4 | 2,5 | 3,1 | 3,1 | 5,3 | 4,2 | 2,5 | 1,4 |
| 1,2 | 5,1 | 2,3 | 3,4 | 4,5 | 4,5 | 1,2 | 5,1 | 3,4 | 2,3 |

Составление расписания матчей по дням

Присвоение расписанию неустановленных значений

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | | 9 | |
| 0,4 | -1 | 0,3 | -1 | 0,5 | -1 | 0,1 | -1 | 0,2 | -1 | 0,2 | -1 | 0,4 | -1 | 0,3 | -1 | 0,1 | -1 | 0,5 | -1 |
| 5,3 | -1 | 4,2 | -1 | 1,4 | -1 | 2,5 | -1 | 3,1 | -1 | 3,1 | -1 | 5,3 | -1 | 4,2 | -1 | 2,5 | -1 | 1,4 | -1 |
| 1,2 | -1 | 5,1 | -1 | 2,3 | -1 | 3,4 | -1 | 4,5 | -1 | 4,5 | -1 | 1,2 | -1 | 5,1 | -1 | 3,4 | -1 | 2,3 | -1 |

Счетчик выполнения предпочтений для дней недели со сбросом раз в 4 дня

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | | 1 | | 2 | | 3 | | 4 | | 5 | |
| 1 | 3 | 1 | 2 | 3 | 2 |  |  |  |  | 1 | 3 |
|  |  | 4 | 2 |  |  |  |  |  |  |  |  |

Счетчик матчей в день

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Выделение 0 туру первые часовые слоты из числа свободных в количестве общего числа свободных слотов деленного на число неразмещенных туров

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Размещение 0 тура по первому попавшемуся неисполненному предпочтению

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | | 9 | |
| 0,4 | 1 | 0,3 | -1 | 0,5 | -1 | 0,1 | -1 | 0,2 | -1 | 0,2 | -1 | 0,4 | -1 | 0,3 | -1 | 0,1 | -1 | 0,5 | -1 |
| 5,3 | -1 | 4,2 | -1 | 1,4 | -1 | 2,5 | -1 | 3,1 | -1 | 3,1 | -1 | 5,3 | -1 | 4,2 | -1 | 2,5 | -1 | 1,4 | -1 |
| 1,2 | -1 | 5,1 | -1 | 2,3 | -1 | 3,4 | -1 | 4,5 | -1 | 4,5 | -1 | 1,2 | -1 | 5,1 | -1 | 3,4 | -1 | 2,3 | -1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | | 1 | | 2 | | 3 | | 4 | | 5 | |
| 1 | 2 | 1 | 2 | 3 | 2 |  |  |  |  | 1 | 3 |
|  |  | 4 | 2 |  |  |  |  |  |  |  |  |

Размещение 0 тура для оставшихся матчей сначала в дни с недостающим числом игр до минимального (кроме последнего дня в туре), а после в дни со свободными часами в исходном порядке.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | | 9 | |
| 0,4 | 1 | 0,3 | -1 | 0,5 | -1 | 0,1 | -1 | 0,2 | -1 | 0,2 | -1 | 0,4 | -1 | 0,3 | -1 | 0,1 | -1 | 0,5 | -1 |
| 5,3 | 0 | 4,2 | -1 | 1,4 | -1 | 2,5 | -1 | 3,1 | -1 | 3,1 | -1 | 5,3 | -1 | 4,2 | -1 | 2,5 | -1 | 1,4 | -1 |
| 1,2 | 0 | 5,1 | -1 | 2,3 | -1 | 3,4 | -1 | 4,5 | -1 | 4,5 | -1 | 1,2 | -1 | 5,1 | -1 | 3,4 | -1 | 2,3 | -1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Перенос матчей с конца в дни (кроме последнего дня в туре) с недостающим числом игр с начала.

…

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | | 9 | |
| 0,4 | 1 | 0,3 | 1 | 0,5 | 2 | 0,1 | 3 | 0,2 | 4 | 0,2 | 5 | 0,4 | 6 | 0,3 | 7 | 0,1 | 9 | 0,5 | 11 |
| 5,3 | 0 | 4,2 | 2 | 1,4 | 2 | 2,5 | 3 | 3,1 | 5 | 3,1 | 6 | 5,3 | 7 | 4,2 | 8 | 2,5 | 8 | 1,4 | 11 |
| 1,2 | 0 | 5,1 | 1 | 2,3 | 3 | 3,4 | 4 | 4,5 | 4 | 4,5 | 6 | 1,2 | 7 | 5,1 | 8 | 3,4 | 9 | 2,3 | 13 |

Составление расписания матчей по часам

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | | | 1 | | | 2 | | | 3 | | | 4 | | |
| 0,4 | 1 | -1 | 0,3 | 1 | -1 | 0,5 | 2 | -1 | 0,1 | 3 | -1 | 0,2 | 4 | -1 |
| 5,3 | 0 | -1 | 4,2 | 2 | -1 | 1,4 | 2 | -1 | 2,5 | 3 | -1 | 3,1 | 5 | -1 |
| 1,2 | 0 | -1 | 5,1 | 1 | -1 | 2,3 | 3 | -1 | 3,4 | 4 | -1 | 4,5 | 4 | -1 |
| 5 | | | 6 | | | 7 | | | 8 | | | 9 | | |
| 0,2 | 5 | -1 | 0,4 | 6 | -1 | 0,3 | 7 | -1 | 0,1 | 9 | -1 | 0,5 | 11 | -1 |
| 3,1 | 6 | -1 | 5,3 | 7 | -1 | 4,2 | 8 | -1 | 2,5 | 8 | -1 | 1,4 | 11 | -1 |
| 4,5 | 6 | -1 | 1,2 | 7 | -1 | 5,1 | 8 | -1 | 3,4 | 9 | -1 | 2,3 | 13 | -1 |

Метки занятых часов

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

… последний алгоритм не дописан