



Mesterséges Intelligencia Alapok

Féléves feladat

Jegyzőkönyv

Készítette:

Hegedűs Attila László
Mérnök-informatikus
levelező
D2OVJ9

Miskolc, 2022

Feladat bemutatása

Flow-shop ütemezés: A flow-shop feladat egy ún. többoperációs gyártásütemezési feladat. A gyártásütemezési feladatok formális leírására az alfa, béta, gamma jelölést alkalmazzuk ($\alpha|\beta|\gamma$). E szerint a rendszer szerint az egyszerű flow-shop feladat leírása:

$$F||C_{\max}$$

Jelentése:

Az erőforrások az ütemezési időszakban folyamatosan rendelkezésre állnak. Az erőforrások egyszerre csak egy munkán dolgoznak. A munkák legkorábbi indítási pontja nulla (bármikor indíthatóak). Minden egyes munkához adott m számú operáció tartozik, melyeknek pontosan ismert a végrehajtási ideje:

$$p_{i,j}, i \in 1, 2, \dots, n, j \in 1, 2, \dots, m.$$

Az operációk végrehajtási sorrendje kötött és minden munka esetében azonos. Az operációk végrehajtása nem szakítható meg. A gépek között a munkák várakozhatnak, a műveletközi tárolók mérete nem korlátos. Az ütemezés célja az utolsóként elkészülő munka befejezési időpontjának minimalizálása.

Megoldás bemutatása

A megoldásom kódját két részre osztottam. A `beadando_final.py` fájl az ütemezést számítja ki, a `gantt.py` fájl a gantt diagram ábrázolásához szükséges kódot tartalmazza. A program futása során létrehoz egy `makespan_log.txt` nevű szöveges fájlt, amiben naplózza az egyes munka végrehajtási sorrendhez tartozó utolsó munka befejezési időpontokat és hogy hányadik iterációnál jár. A `best_sequence_log.txt` file tartalmazza a legjobb időhöz tartozó ütemezés részleteit. A `report.txt` fájl tartalmazza a generált operációk idejének mátrixát, a kezdeti befejezési időt és sorrendet, valamint a legjobb időt és végrehajtási sorrendet, amit megtalált.

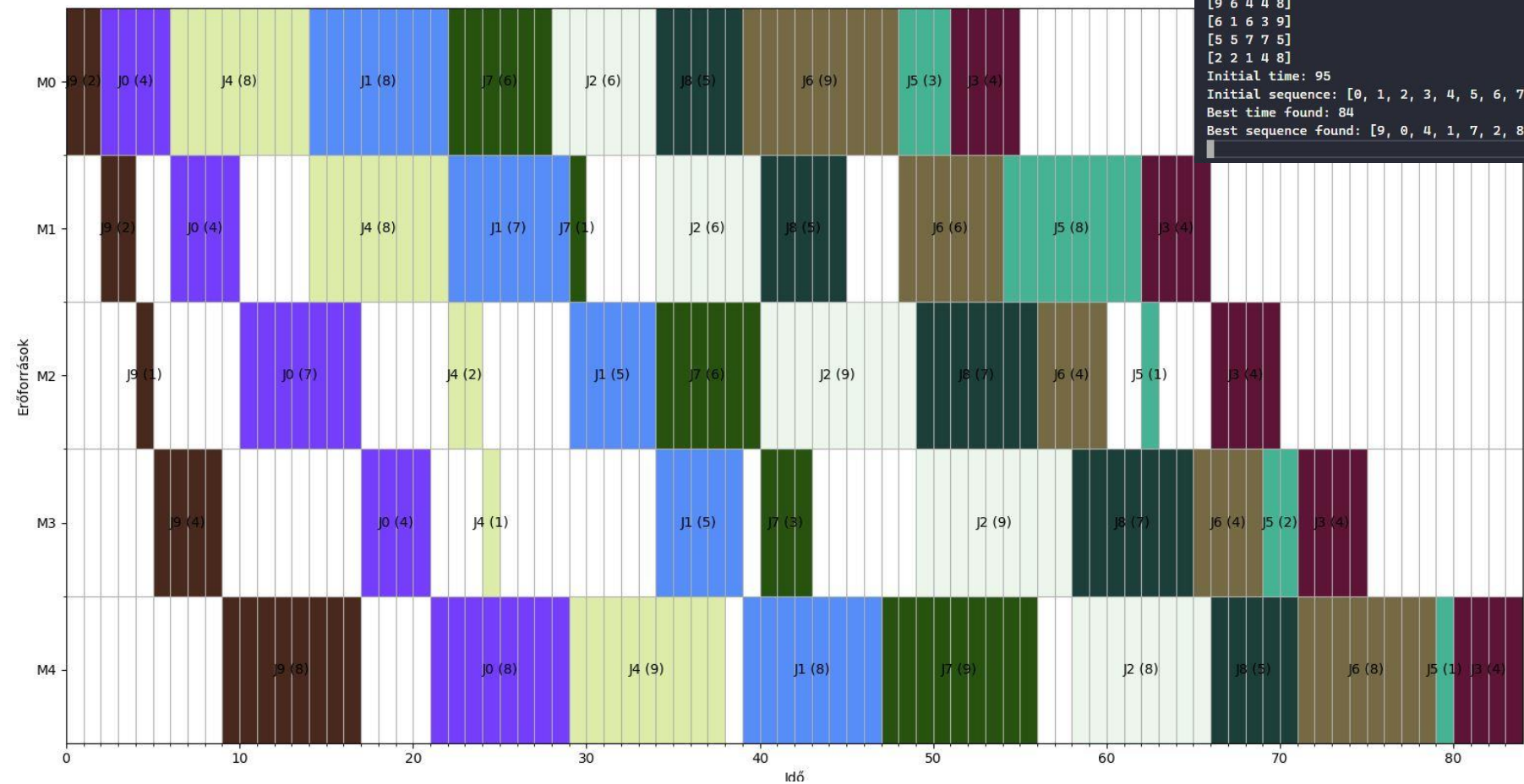
A feladathoz először beállítjuk a kiindulási adatokat, a munkák és erőforrások számát. Ebből a `numpy randint()` függvényének segítségével generáljuk le az operációk idejének mátrixát. A véletlen számokat itt 1 és 10 között generáltam. Megadok továbbá egy kezdeti végrehajtási sorrendet, a hűtés hőfokát, és a kilépési feltételt az iterációk számát. Ezt követően kiszámoljuk a kezdeti végrehajtási sorrendhez tartozó végrehajtási időt a `makespan()` függvénnyel, ez lesz a szimuláció alapja. Ezen adatokat felhasználva indítjuk a szimulációt a `sim_anneal()` nevű függvény felhasználásával. A függvény először beállítja a változóit, ezek tartalmazzák az előző végrehajtási sorrendet, a legjobb időhöz tartozó végrehajtási sorrendet, az előzőnek számolt időt, és a legjobb eddigi időt. Ezután a függvény while loopban (a megadott iterációk számától függ hányszor fut le) generál egy új sorrendet a kezdeti sorrend alapján, majd kiszámolja az új sorrendhez tartozó végrehajtási időt. Ha ez az idő rövidebb, mint az eddigi legjobb átállítja a legjobb időhöz és sorrendhez tartozó változókat, és ebből a sorrendből készíti az új sorrendet. Ha az idő nem jobb, akkor p eséllyel állítjuk be a rosszabb időhöz tartozó sorrendet az új sorrend generálásához. Ezt a p esélyt az `annealing()` függvény szolgáltatja. Iteráció közben a program naplóz egy külön fájlba.

Az iterációnak vége a kilépési feltétel teljesülése után. Ezzel megkapjuk a legjobb időt és hozzá tartozó sorrendet. Ahhoz, hogy gantt diagramot tudjunk rajzolni részletesebb információkra lesz szükségünk. Ezeket a részletes információkat a `get_detailed_schedule()` függvény adja meg nekünk. A függvény egy tömbbe fogja létrehozni az információkat. Ehhez hívja a `get_detailed_subtask()` függvényt, mely egy szótár bejegyzésbe írja bele minden adott részfeladathoz tartozó munka kezdéséhez tartozó időt, az operáció hosszát, az erőforrás azonosítóját és a munka azonosítóját. Ezt a szótár bejegyzést fűzzük hozzá az előző függvény tömbjébe. A tömböt a `gantt_megrajzol()` függvénynek adjuk át, amely elkészíti a diagramot. A függvényhez kell még egy lista az erőforrások neveiről és egy a munkák neveiről az ábra feliratozásához. Ez a függvény már a `gantt.py` fileból lett importálva. A rajzoláshoz a `matplotlib.pyplot` könyvtárat használtam. Az `init_gantt()` függvény beállítja az ábra típusára, kiosztására, feliratozására vonatkozó részleteket. A `reszfeladat_szamitas()` függvény az ábra feltöltését szolgáltatja a részletes munkaterv alapján. A `gantt_befejez()` függvény adja át az információkat a munkatervből a `reszfeladat_szamitas()` függvénynek. A `gantt_letrehoz()` függvény kezdi el az előző függvények meghívását. A `gantt_megrajzol()` függvény meghívja a `gantt_letrehoz()` függvényt és a `matplotlib show()` függvényével megkapjuk az ábra nézetét.

Tesztek

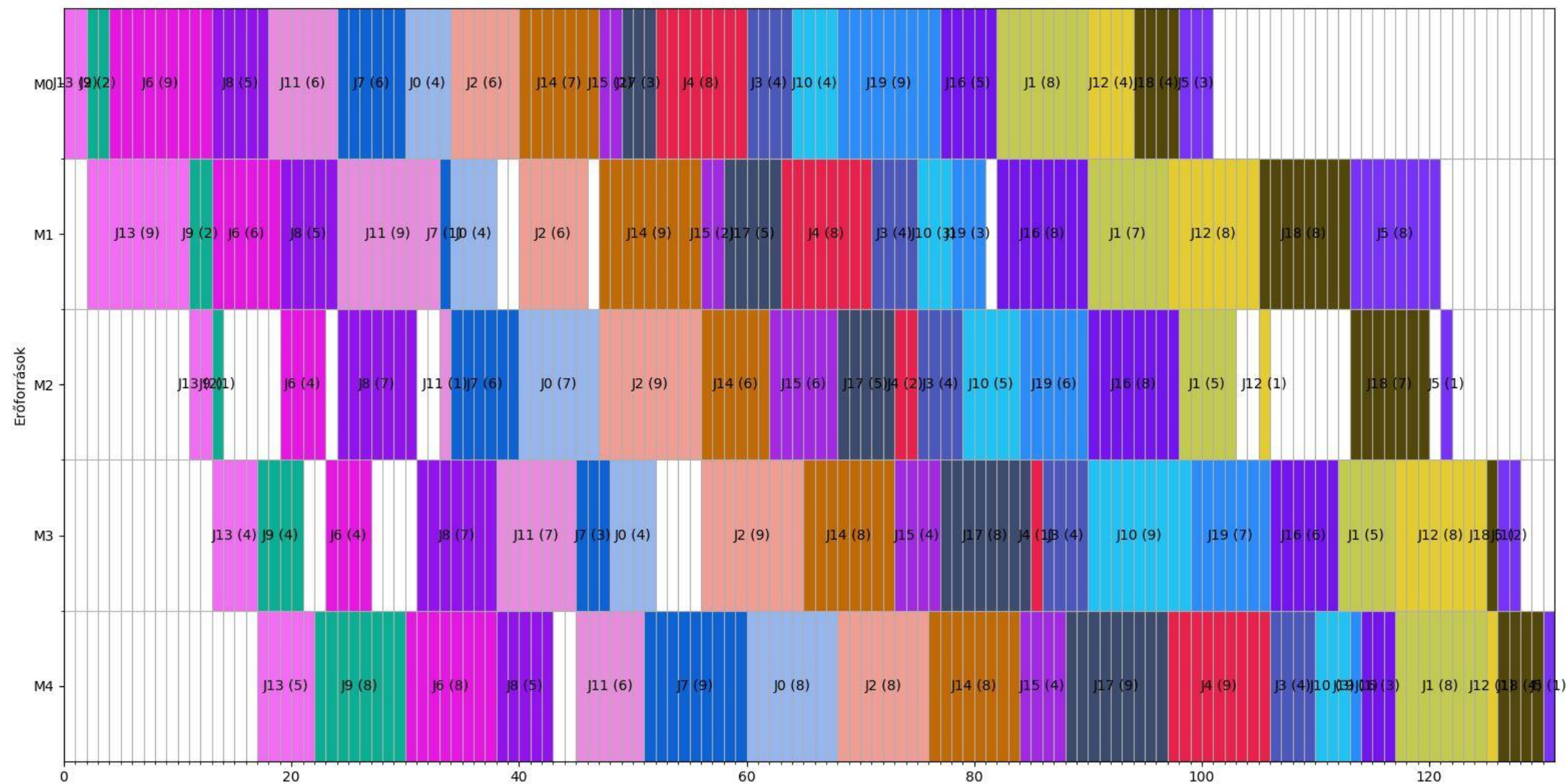
A teszteket 10000 iterációig futtattam, a képek a `report.txt` fájlt és a generált gantt diagramot ábrázolják.

5 gép, 10 munka



```
Array of Jobs:  
[4 4 7 4 8]  
[8 7 5 5 8]  
[6 6 9 9 8]  
[4 4 4 4 4]  
[8 8 2 1 9]  
[3 8 1 2 1]  
[9 6 4 4 8]  
[6 1 6 3 9]  
[5 5 7 7 5]  
[2 2 1 4 8]  
Initial time: 95  
Initial sequence: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]  
Best time found: 84  
Best sequence found: [9, 0, 4, 1, 7, 2, 8, 6, 5, 3]
```

5 gép 20 munka



Array of Jobs:

[4 4 7 4 8]

[8 7 5 5 8]

[6 6 9 9 8]

[4 4 4 4 4]

[8 8 2 1 9]

[3 8 1 2 1]

[9 6 4 4 8]

[6 1 6 3 9]

[5 5 7 7 5]

[2 2 1 4 8]

[4 3 5 9 3]

[6 9 1 7 6]

[4 8 1 8 1]

[2 9 2 4 5]

[7 9 6 8 8]

[2 2 6 4 4]

[5 8 8 6 3]

[3 5 5 8 9]

[4 8 7 1 4]

[9 3 6 7 1]

Initial time: 148

Initial sequence: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]

Best time found: 131

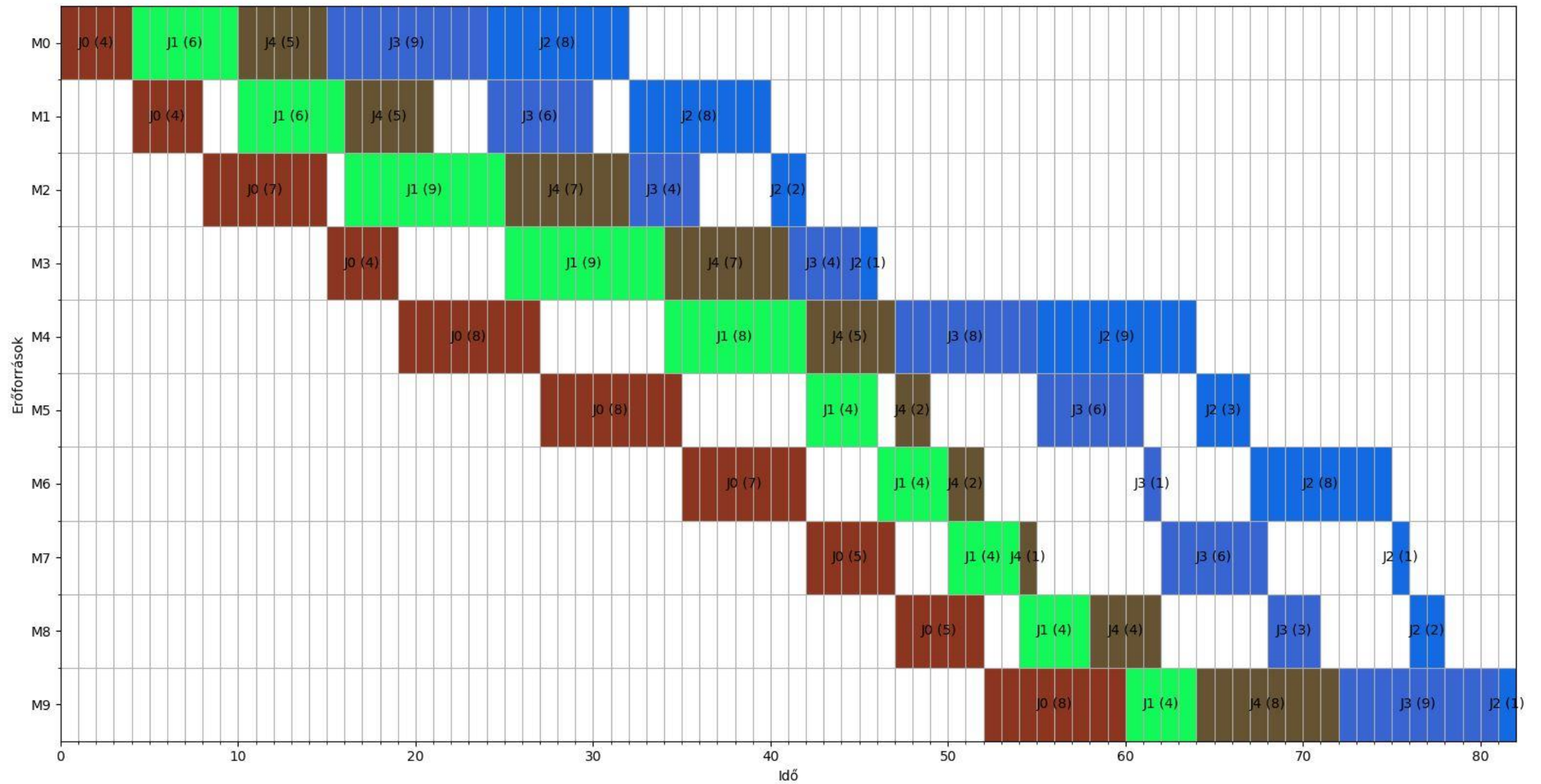
Best sequence found: [13, 9, 6, 8, 11, 7, 0, 2, 14, 15, 17, 4, 3, 10, 19, 16, 1, 12, 18, 5]

5 gép 50 munka (a diagram olvashatatlan)

Array of Jobs:

```
[4 4 7 4 8]
[8 7 5 5 8]
[6 6 9 9 8]
[4 4 4 4 4]
[8 8 2 1 9]
[3 8 1 2 1]
[9 6 4 4 8]
[6 1 6 3 9]
[5 5 7 7 5]
[2 2 1 4 8]
[4 3 5 9 3]
[6 9 1 7 6]
[4 8 1 8 1]
[2 9 2 4 5]
[7 9 6 8 8]
[2 2 6 4 4]
[5 8 8 6 3]
[3 5 5 8 9]
[4 8 7 1 4]
[9 3 6 7 1]
[5 5 2 3 3]
[9 4 5 9 6]
[5 2 2 7 6]
[3 9 4 2 9]
[8 2 5 8 5]
[2 2 6 6 2]
[7 7 6 8 4]
[5 1 4 4 7]
[3 7 5 3 2]
[6 8 3 8 9]
[5 6 6 7 6]
[6 1 3 3 4]
[3 5 4 3 8]
[6 6 5 6 2]
[4 2 3 3 2]
[5 3 5 4 9]
[9 6 2 6 4]
[9 7 5 3 1]
[3 4 1 3 8]
[4 9 1 7 9]
[9 2 5 3 1]
[4 8 5 3 3]
[4 8 9 2 1]
[2 9 1 3 6]
[3 6 5 7 9]
[9 4 5 3 2]
[1 9 4 6 3]
[4 3 1 2 5]
[6 7 4 3 4]
[9 1 6 2 3]
Initial time: 302
Initial sequence: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49]
Best time found: 280
Best sequence found: [9, 43, 29, 3, 8, 46, 13, 27, 7, 6, 35, 42, 39, 48, 25, 14, 44, 10, 49, 5, 47, 22, 4, 32, 24, 23, 31, 2, 19, 12, 0, 18, 40, 34, 15, 30, 17, 26, 38, 16, 45, 11, 41, 21, 1, 20, 36, 37, 33, 28]
```

10 gép 5 munka



Array of Jobs:

[4 4 7 4 8 8 7 5 5 8]

[6 6 9 9 8 4 4 4 4 4]

[8 8 2 1 9 3 8 1 2 1]

[9 6 4 4 8 6 1 6 3 9]

[5 5 7 7 5 2 2 1 4 8]

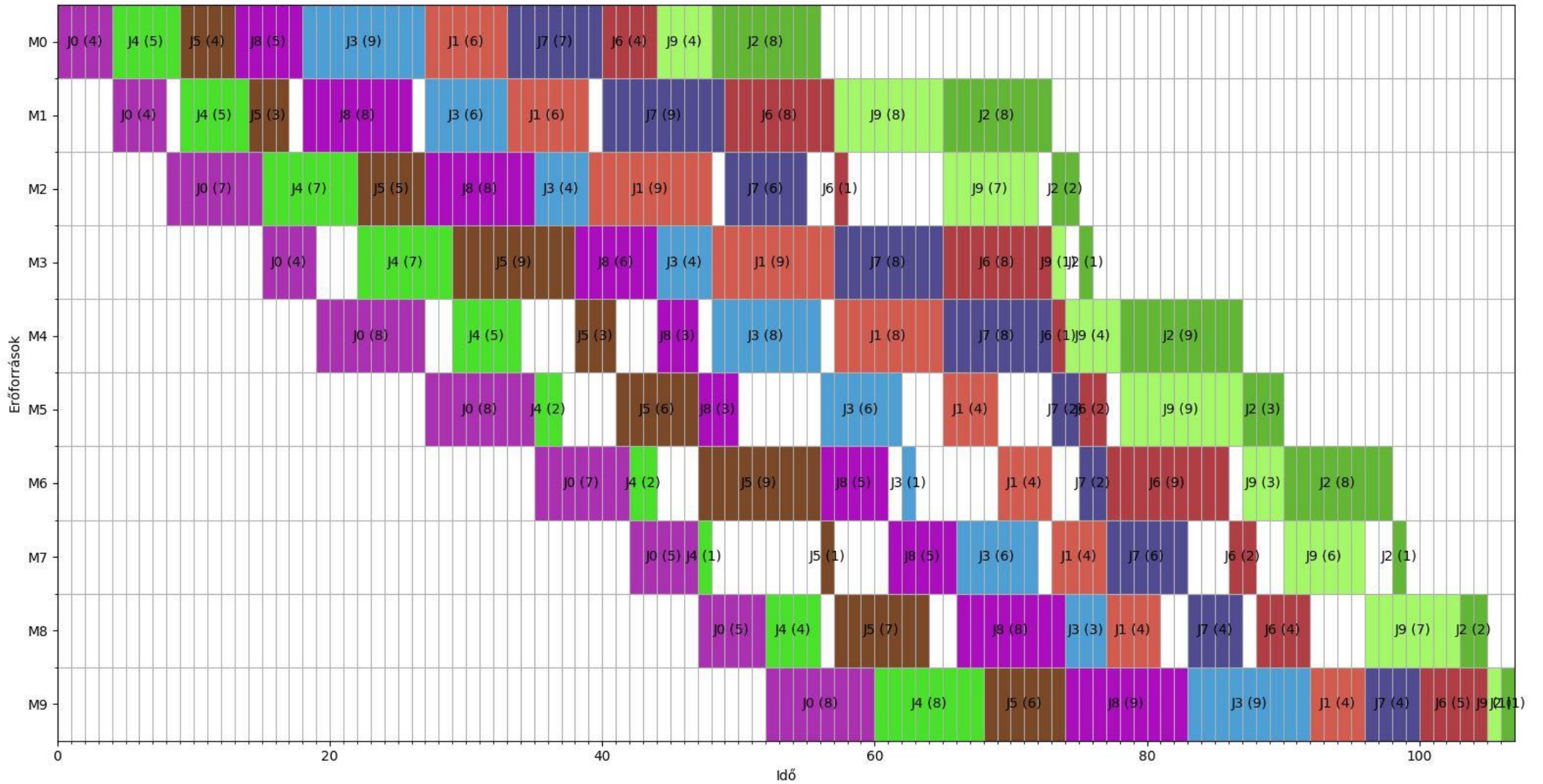
Initial time: 92

Initial sequence: [0, 1, 2, 3, 4]

Best time found: 82

Best sequence found: [0, 1, 4, 3, 2]

10 gép 10 munka



Array of Jobs:

[4 4 7 4 8 8 7 5 5 8]

[6 6 9 9 8 4 4 4 4 4]

[8 8 2 1 9 3 8 1 2 1]

[9 6 4 4 8 6 1 6 3 9]

[5 5 7 7 5 2 2 1 4 8]

[4 3 5 9 3 6 9 1 7 6]

[4 8 1 8 1 2 9 2 4 5]

[7 9 6 8 8 2 2 6 4 4]

[5 8 8 6 3 3 5 5 8 9]

[4 8 7 1 4 9 3 6 7 1]

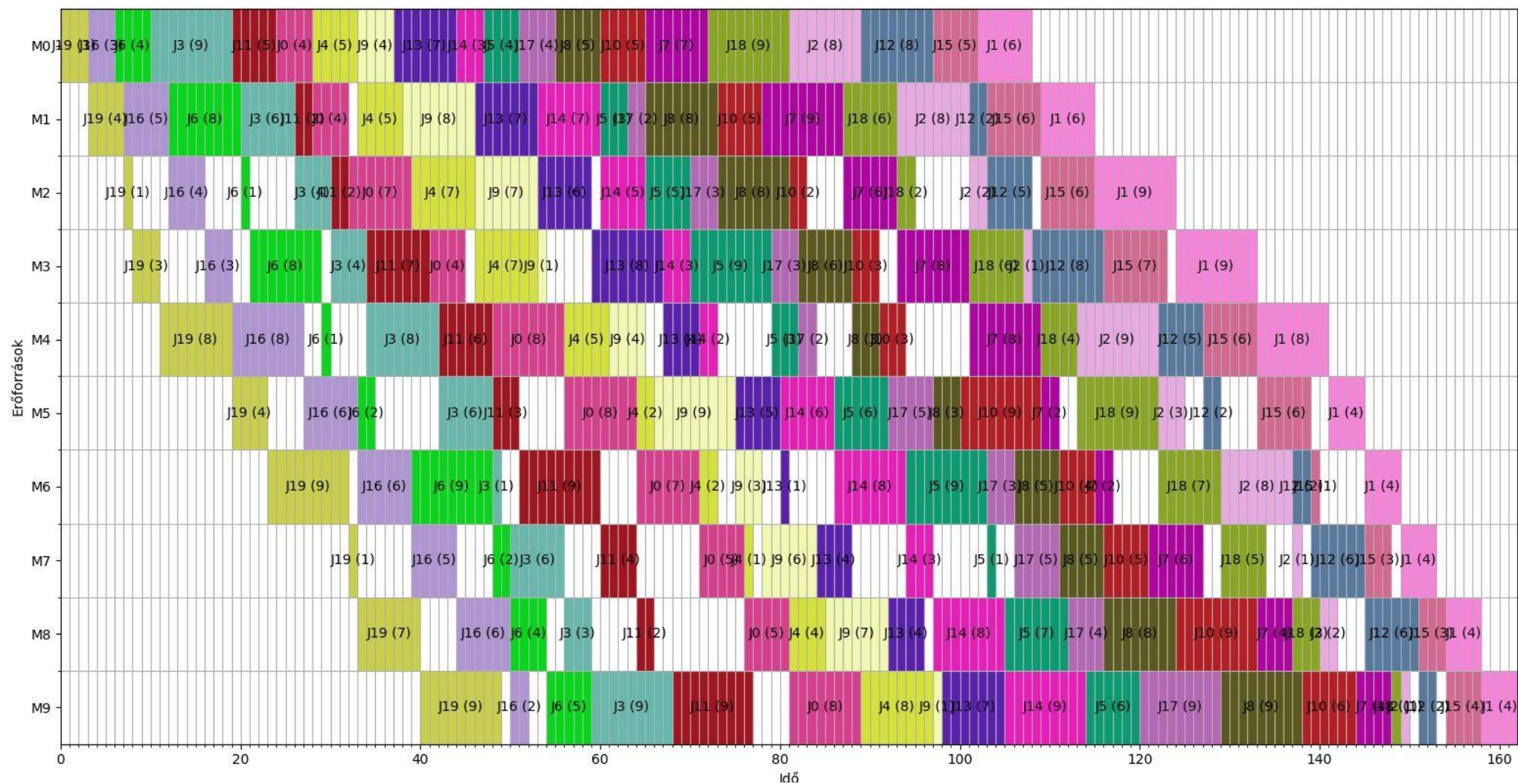
Initial time: 122

Initial sequence: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

Best time found: 107

Best sequence found: [0, 4, 5, 8, 3, 1, 7, 6, 9, 2]

10 gép, 20 munka



Array of Jobs:

```
[4 4 7 4 8 8 7 5 5 8]
[6 6 9 9 8 4 4 4 4 4]
[8 8 2 1 9 3 8 1 2 1]
[9 6 4 4 8 6 1 6 3 9]
[5 5 7 7 5 2 2 1 4 8]
[4 3 5 9 3 6 9 1 7 6]
[4 8 1 8 1 2 9 2 4 5]
[7 9 6 8 8 2 2 6 4 4]
[5 8 8 6 3 3 5 5 8 9]
[4 8 7 1 4 9 3 6 7 1]
[5 5 2 3 3 9 4 5 9 6]
[5 2 2 7 6 3 9 4 2 9]
[8 2 5 8 5 2 2 6 6 2]
[7 7 6 8 4 5 1 4 4 7]
[3 7 5 3 2 6 8 3 8 9]
[5 6 6 7 6 6 1 3 3 4]
[3 5 4 3 8 6 6 5 6 2]
[4 2 3 3 2 5 3 5 4 9]
[9 6 2 6 4 9 7 5 3 1]
[3 4 1 3 8 4 9 1 7 9]
```

Initial time: 187

Initial sequence: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]

Best time found: 162

Best sequence found: [19, 16, 6, 3, 11, 0, 4, 9, 13, 14, 5, 17, 8, 10, 7, 18, 2, 12, 15, 1]

10 gép 50 munka (a diagram olvashatatlan)

Array of Jobs:

```
[4 4 7 4 8 8 7 5 5 8]
[6 6 9 9 8 4 4 4 4 4]
[8 8 2 1 9 3 8 1 2 1]
[9 6 4 4 8 6 1 6 3 9]
[5 5 7 7 5 2 2 1 4 8]
[4 3 5 9 3 6 9 1 7 6]
[4 8 1 8 1 2 9 2 4 5]
[7 9 6 8 8 2 2 6 4 4]
[5 8 8 6 3 3 5 5 8 9]
[4 8 7 1 4 9 3 6 7 1]
[5 5 2 3 3 9 4 5 9 6]
[5 2 2 7 6 3 9 4 2 9]
[8 2 5 8 5 2 2 6 6 2]
[7 7 6 8 4 5 1 4 4 7]
[3 7 5 3 2 6 8 3 8 9]
[5 6 6 7 6 6 1 3 3 4]
[3 5 4 3 8 6 6 5 6 2]
[4 2 3 3 2 5 3 5 4 9]
[9 6 2 6 4 9 7 5 3 1]
[3 4 1 3 8 4 9 1 7 9]
[9 2 5 3 1 4 8 5 3 3]
[4 8 9 2 1 2 9 1 3 6]
[3 6 5 7 9 9 4 5 3 2]
[1 9 4 6 3 4 3 1 2 5]
[6 7 4 3 4 9 1 6 2 3]
[3 2 5 2 9 7 2 8 1 5]
[3 3 9 8 3 8 6 9 4 8]
[6 9 3 4 4 7 2 4 8 6]
[5 8 9 6 5 6 9 3 5 7]
[2 2 7 4 6 3 2 9 7 7]
[9 5 5 9 2 2 6 3 5 8]
[4 6 5 9 2 7 1 1 9 7]
[5 1 4 9 7 8 1 2 5 9]
[7 9 5 8 7 3 7 8 3 1]
[5 7 2 4 4 5 8 4 3 9]
[8 3 5 7 2 7 3 8 7 9]
[3 4 5 4 9 9 6 1 6 7]
[5 1 9 5 7 4 5 5 7 1]
[9 1 5 3 6 7 2 6 9 8]
[7 4 8 5 3 3 5 8 3 9]
[5 4 4 7 1 3 8 2 5 1]
[4 7 6 6 7 1 1 4 8 5]
[4 3 9 9 7 7 4 4 3 7]
[8 6 9 8 5 6 3 6 5 3]
[7 6 1 2 2 9 7 4 1 6]
[8 6 4 1 3 2 2 7 5 6]
[6 2 7 7 7 5 4 1 4 7]
[2 6 2 8 9 2 2 5 7 7]
[9 8 8 4 5 6 9 5 2 3]
[4 8 1 7 2 3 9 6 3 1]
```

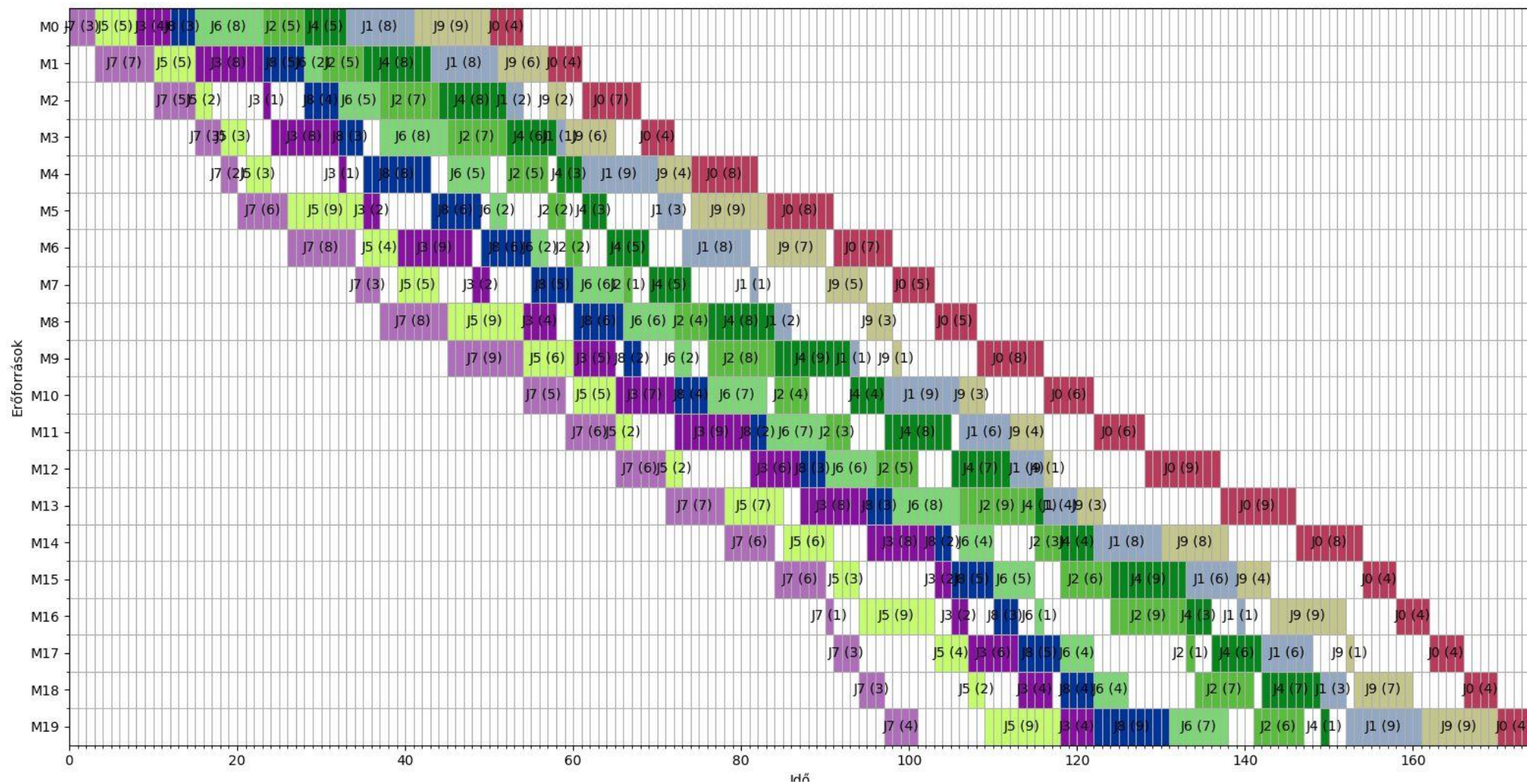
Initial time: 366

Initial sequence: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49]

Best time found: 341

Best sequence found: [29, 14, 16, 10, 32, 40, 5, 39, 45, 22, 23, 30, 36, 47, 0, 21, 13, 2, 12, 31, 11, 24, 9, 19, 38, 42, 1, 8, 34, 33, 43, 41, 44, 3, 35, 18, 17, 46, 26, 20, 48, 4, 28, 7, 27, 6, 37, 25, 49, 15]

20 gép 10 munka



Array of Jobs:

```
[4 4 7 4 8 8 7 5 5 8 6 6 9 9 8 4 4 4 4 4]
[8 8 2 1 9 3 8 1 2 1 9 6 4 4 8 6 1 6 3 9]
[5 5 7 7 5 2 2 1 4 8 4 3 5 9 3 6 9 1 7 6]
[4 8 1 8 1 2 9 2 4 5 7 9 6 8 8 2 2 6 4 4]
[5 8 8 6 3 3 5 5 8 9 4 8 7 1 4 9 3 6 7 1]
[5 5 2 3 3 9 4 5 9 6 5 2 2 7 6 3 9 4 2 9]
[8 2 5 8 5 2 2 6 6 2 7 7 6 8 4 5 1 4 4 7]
[3 7 5 3 2 6 8 3 8 9 5 6 6 7 6 6 1 3 3 4]
[3 5 4 3 8 6 6 5 6 2 4 2 3 3 2 5 3 5 4 9]
[9 6 2 6 4 9 7 5 3 1 3 4 1 3 8 4 9 1 7 9]
```

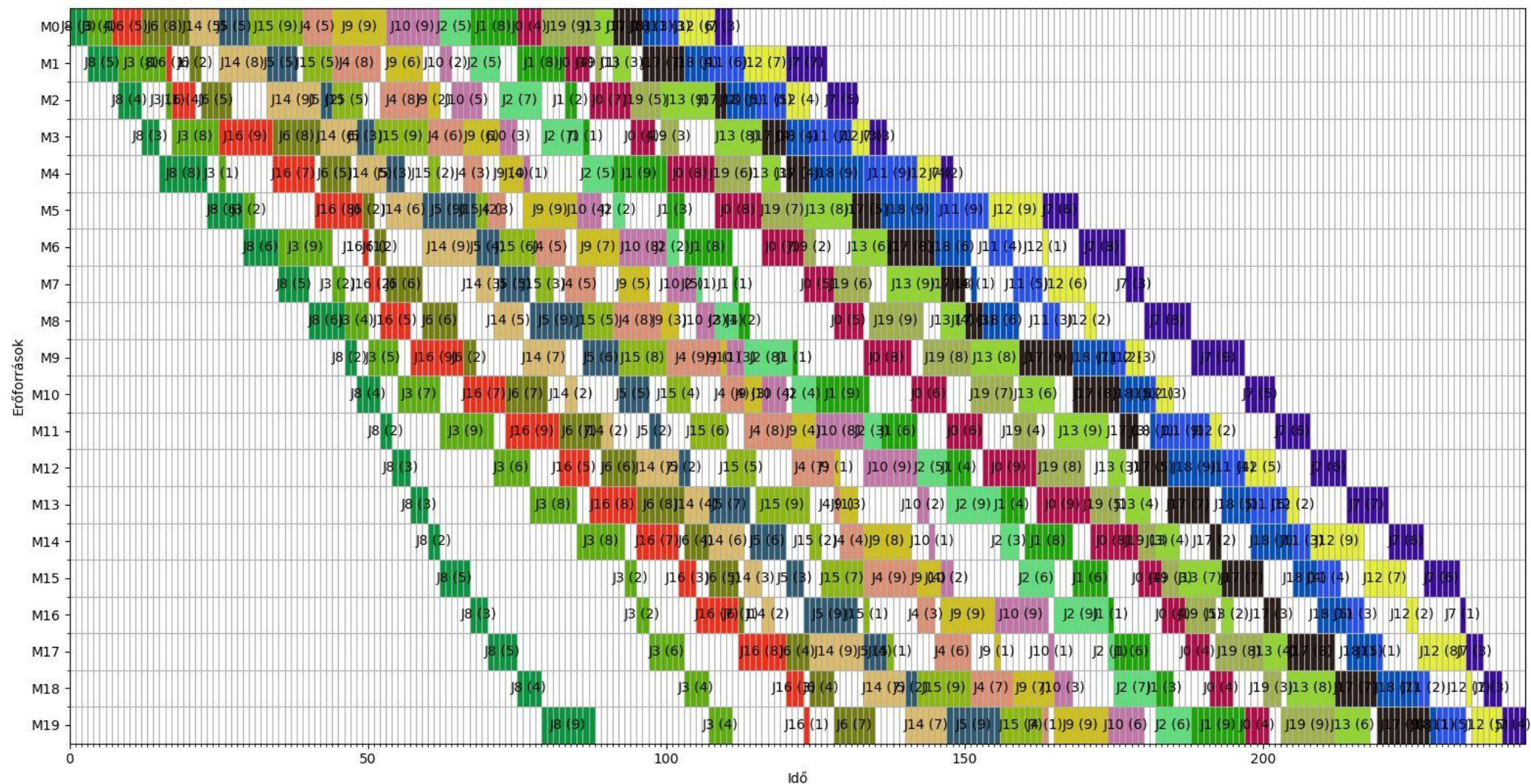
Initial time: 188

Initial sequence: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

Best time found: 174

Best sequence found: [7, 5, 3, 8, 6, 2, 4, 1, 9, 0]

20 gép 20 munka



Array of Jobs:

```
[4 4 7 4 8 8 7 5 5 8 6 6 9 9 8 4 4 4 4 4]
[8 8 2 1 9 3 8 1 2 1 9 6 4 4 8 6 1 6 3 9]
[5 5 7 7 5 2 2 1 4 8 4 3 5 9 3 6 9 1 7 6]
[4 8 1 8 1 2 9 2 4 5 7 9 6 8 8 2 2 6 4 4]
[5 8 8 6 3 3 5 5 8 9 4 8 7 1 4 9 3 6 7 1]
[5 5 2 3 3 9 4 5 9 6 5 2 2 7 6 3 9 4 2 9]
[8 2 5 8 5 2 2 6 6 2 7 7 6 8 4 5 1 4 4 7]
[3 7 5 3 2 6 8 3 8 9 5 6 6 7 6 6 1 3 3 4]
[3 5 4 3 8 6 6 5 6 2 4 2 3 3 2 5 3 5 4 9]
[9 6 2 6 4 9 7 5 3 1 3 4 1 3 8 4 9 1 7 9]
[9 2 5 3 1 4 8 5 3 3 4 8 9 2 1 2 9 1 3 6]
[3 6 5 7 9 9 4 5 3 2 1 9 4 6 3 4 3 1 2 5]
[6 7 4 3 4 9 1 6 2 3 3 2 5 2 9 7 2 8 1 5]
[3 3 9 8 3 8 6 9 4 8 6 9 3 4 4 7 2 4 8 6]
[5 8 9 6 5 6 9 3 5 7 2 2 7 4 6 3 2 9 7 7]
[9 5 5 9 2 2 6 3 5 8 4 6 5 9 2 7 1 1 9 7]
[5 1 4 9 7 8 1 2 5 9 7 9 5 8 7 3 7 8 3 1]
[5 7 2 4 4 5 8 4 3 9 8 3 5 7 2 7 3 8 7 9]
[3 4 5 4 9 9 6 1 6 7 5 1 9 5 7 4 5 5 7 1]
[9 1 5 3 6 7 2 6 9 8 7 4 8 5 3 3 5 8 3 9]
```

Initial time: 265

Initial sequence: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]

Best time found: 244

Best sequence found: [8, 3, 16, 6, 14, 5, 15, 4, 9, 10, 2, 1, 0, 19, 13, 17, 18, 11, 12, 7]

20 gép, 50 munka (a diagram ennyi elemmel már olvashatatlan)

```
Array of Jobs:
[4 4 7 4 8 8 7 5 5 8 6 6 9 9 8 4 4 4 4 4]
[8 8 2 1 9 3 8 1 2 1 9 6 4 4 8 6 1 6 3 9]
[5 5 7 7 5 2 2 1 4 8 4 3 5 9 3 6 9 1 7 6]
[4 8 1 8 1 2 9 2 4 5 7 9 6 8 8 2 2 6 4 4]
[5 8 8 6 3 3 5 5 8 9 4 8 7 1 4 9 3 6 7 1]
[5 5 2 3 3 9 4 5 9 6 5 2 2 7 6 3 9 4 2 9]
[8 2 5 8 5 2 2 6 6 2 7 7 6 8 4 5 1 4 4 7]
[3 7 5 3 2 6 8 3 8 9 5 6 6 7 6 6 1 3 3 4]
[3 5 4 3 8 6 6 5 6 2 4 2 3 3 2 5 3 5 4 9]
[9 6 2 6 4 9 7 5 3 1 3 4 1 3 8 4 9 1 7 9]
[9 2 5 3 1 4 8 5 3 3 4 8 9 2 1 2 9 1 3 6]
[3 6 5 7 9 9 4 5 3 2 1 9 4 6 3 4 3 1 2 5]
[6 7 4 3 4 9 1 6 2 3 3 2 5 2 9 7 2 8 1 5]
[3 3 9 8 3 8 6 9 4 8 6 9 3 4 4 7 2 4 8 6]
[5 8 9 6 5 6 9 3 5 7 2 2 7 4 6 3 2 9 7 7]
[9 5 5 9 2 2 6 3 5 8 4 6 5 9 2 7 1 1 9 7]
[5 1 4 9 7 8 1 2 5 9 7 9 5 8 7 3 7 8 3 1]
[5 7 2 4 4 5 8 4 3 9 8 3 5 7 2 7 3 8 7 9]
[3 4 5 4 9 9 6 1 6 7 5 1 9 5 7 4 5 5 7 1]
[9 1 5 3 6 7 2 6 9 8 7 4 8 5 3 3 5 8 3 9]
[5 4 4 7 1 3 8 2 5 1 4 7 6 6 7 1 1 4 8 5]
[4 3 9 9 7 7 4 4 3 7 8 6 9 8 5 6 3 6 5 3]
[7 6 1 2 2 9 7 4 1 6 8 6 4 1 3 2 2 7 5 6]
[6 2 7 7 7 5 4 1 4 7 2 6 2 8 9 2 2 5 7 7]
[9 8 8 4 5 6 9 5 2 3 4 8 1 7 2 3 9 6 3 1]
[1 8 4 3 9 4 4 8 4 1 4 1 8 4 7 7 8 8 4 3]
[1 9 8 4 8 3 5 7 2 5 4 4 6 3 4 6 4 4 9 1]
[7 3 7 2 3 3 6 1 4 6 7 1 4 7 2 1 9 5 7 7]
[6 9 5 5 6 8 6 2 7 8 5 8 8 2 3 3 8 4 5 8]
[2 7 7 8 8 3 4 4 1 9 5 4 9 1 5 8 7 6 9 7]
[7 7 5 1 9 9 1 7 3 8 5 4 4 2 1 9 7 2 3 7]
[8 9 3 7 3 2 7 5 1 3 3 7 6 3 9 8 2 1 9 8]
[1 5 3 5 6 4 6 2 8 3 5 2 3 2 2 6 3 2 8 2]
[4 9 7 9 8 3 4 2 1 4 1 2 1 8 1 9 9 2 9 2]
[2 7 6 9 1 9 1 8 9 6 7 6 8 1 2 9 7 1 9 4]
[2 9 5 6 7 1 9 3 4 8 9 2 1 4 8 4 4 4 9 2]
[1 4 4 1 3 8 5 7 9 4 4 6 2 1 3 9 3 4 6 4]
[3 1 8 1 9 3 3 9 1 7 2 8 4 5 4 1 2 4 5 3]
[5 7 7 6 6 1 5 2 5 1 6 7 5 6 1 7 7 7 6 2]
[8 4 3 7 6 6 5 8 9 3 7 2 9 2 4 8 2 3 6 8]
[9 6 9 6 2 7 7 9 6 6 5 3 4 3 9 5 5 7 3 9]
[1 7 3 2 5 9 9 7 4 9 1 2 4 8 2 5 3 4 4 1]
[3 7 8 8 6 2 9 3 3 5 3 1 9 8 2 3 8 6 9 6]
[1 1 2 9 8 3 3 3 6 7 6 2 6 9 7 2 5 5 4 8]
[3 2 6 9 4 3 3 9 7 1 4 3 7 4 8 3 5 8 8 2]
[1 9 2 5 2 1 3 1 2 3 7 4 6 7 7 9 8 1 2 6]
[9 9 6 5 3 2 9 5 1 2 7 5 8 6 6 8 8 1 6 9]
[4 6 1 5 4 3 4 9 2 8 9 5 9 6 5 3 3 5 9 6]
[3 3 3 9 2 9 1 4 9 1 2 4 7 1 1 9 8 4 7 7]
[3 4 7 1 1 6 4 1 9 8 4 4 1 8 4 1 1 8 3 3]
Initial time: 463
Initial sequence: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49]
Best time found: 431
Best sequence found: [48, 6, 21, 13, 17, 5, 33, 42, 29, 15, 36, 10, 1, 37, 31, 34, 3, 30, 46, 23, 22, 2, 27, 0, 32, 18, 8, 44, 38, 19, 26, 7, 20, 16, 9, 25, 35, 39, 45, 11, 14, 12, 43, 47, 28, 40, 4, 49, 41, 24]
```