School of Computer Sciences, USM, Penang

CPT244: Artificial Intelligence

Academic Session: Semester 2, 2019/2020

Assignment 02

Presentation Scheduling Using Genetic Algorithm or/and Hybrid System

Presentation is a commonly held activity in a tertiary academic institution. When a presentation is held, it involves many different people. The people gather at a venue, for a specific period such that the speaker will explain a topic of common interest in front of the audiences.

Presentation scheduling is an activity to create a schedule for a set of presentations in the chronological order in which such things are intended to take place. In this project, you are required to produce a scheduler using Genetic Algorithm or/and Hybrid System. The scheduler is able to generate a schedule for a set of 118 presentations with the following requirements:

1. Each presentation (i.e. P1 to P118) consists of a speaker and three audiences (i.e. supervisor and two examiners). Please refer to `SupExaAssign.csv`.
2. There are altogether 47 academic staffs (i.e. S001 to S047) who will play the role either as supervisor or examiner.
3. There are altogether 4 different venues Viva Room (VR), Meeting Room (MR), Interaction Room (IR), and BJIM Discussion Room (BJIM).
4. All the 118 presentations must be assigned to a time slot (no sharing of venue), at a specific venue as shown in the following figure. The time slots are labelled as follows:

Day	Venue	0900 - 0930	0930 - 1000	1000 - 1030	1030 - 1100	1100 - 1130	1130 - 1200	1200 - 1230	1230 - 1300	1400 - 1430	1430 - 1500	1500 - 1530	1530 - 1600	1600 - 1630	1630 - 1700	1700 - 1730
Monday	Viva Room, Level 7	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	Meeting Room, Level 7	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
	Interaction Room, Level 7	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
	BJIM Discussion Room, Level 5	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
Tuesday	Viva Room, Level 7	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75
	Meeting Room, Level 7	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90
	Interaction Room, Level 7	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105
	BJIM Discussion Room, Level 5	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120
Wednesday	Viva Room, Level 7	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135
	Meeting Room, Level 7	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150
	Interaction Room, Level 7	151	152	153	154	155	156	157	158	159	160	161	162	163	164	165
	BJIM Discussion Room, Level 5	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180
Thursday	Viva Room, Level 7	181	182	183	184	185	186	187	188	189	190	191	192	193	194	195
	Meeting Room, Level 7	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210
	Interaction Room, Level 7	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225
	BJIM Discussion Room, Level 5	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240
Friday	Viva Room, Level 7	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255
	Meeting Room, Level 7	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270
	Interaction Room, Level 7	271	272	273	274	275	276	277	278	279	280	281	282	283	284	285
	BJIM Discussion Room, Level 5	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300

5. **Constraint HC01**: All the presentations must be scheduled and each presentation must be scheduled only once.
6. **Constraint HC02**: No staff can attend two or more presentations concurrently.
7. **Constraint HC03** (venue unavailability) – Some venues are not available on a specific time slot. Please refer to `HC03.csv`.
8. **Constraint HC04** (staff unavailability) – Some staffs are not available on a specific time slot. Please refer to `HC04.csv`.
9. **Constraint SC01** (consecutive presentations) – Some staffs prefer attending a few back-to-back presentations (i.e. consecutive presentations). Staffs' consecutive presentations preferences are expressed in `SC01.csv`. For example, if it is stated that a staff's preference is 3, it means that the staff prefers to attend at most three consecutives presentations in a row.
10. **Constraint SC02** (number of days) – Some staffs prefer attending all of his/her presentation within a certain number of days. Staffs' preferences on the number of days in a presentation week are expressed in `SC02.csv`. For example, if it is stated that a staff's preference is 2, it means that the staff prefers to attend all of his/her assigned presentations within two days (may or may not be consecutive).
11. **Constraint SC03** (change of venue) – Some staffs prefers attending all of his/her consecutive presentations at the same venue. Hence, change of venue for the consecutive presentations should be minimized. Staffs' preferences on the change of venue are expressed in `SC03.csv`. For example, if it is stated that a staff's preference is 'y', it means that the staff prefers attending his/her consecutive presentations without changing the venue. Otherwise, if it is stated as 'n', the staff can tolerate change of venue for the consecutive presentations.
12. Constraints **HC01, HC02, HC03, and HC04 are hard constraints**; any violation of these constraints would make the generated schedule an infeasible schedule.
13. Constraints **SC01, SC02 and SC03 are soft constraints**; any violation of these constraints would not make the generated schedule an infeasible schedule. However, the generated schedule must fulfill as many of these constraints as possible.

A generated schedule will be evaluated using a penalty function. This penalty function will give penalty points of 1000 if any of the hard constraint is violated. On the other hand, 10 penalty points will be given a schedule which violates any of the soft constraint. A schedule with low penalty points is preferred. **You may have a different penalty scheme for the soft constraints only**. The penalty scheme has to be detailed in the report.

It is **optional** to optimize constraint SC03. However, bonus grade will be given if constraint SC03 is successfully optimized.

The generated presentation schedule must be in csv format. For example, a fragment of the generated presentation schedule is as follows:

```
P1,null,null,P115,P112,null,P83,P36,…
```

The above example of the fragmented schedule denotes that presentations 1, 115, 112, 83, and 36 are scheduled in slots 1, 4, 5, 7, and 8 respectively; whereas no presentation is scheduled in slots 2, 3, and 6.

This project can be implemented using any programming language of your own choice (C++, JAVA, Python, etc).

Note that the domain problem and the constraints are the **same**. However, the students are encouraged to add additional requirement that leads to the use of any other algorithm that can be hybridized with your main algorithm (additional marks will be given).

Project Due Date: 19 June 2020 (Friday), 11:59p.m.

Assignment Type: Pair assignment.

Submission Procedure: A zip/rar package which consists of the following items is to be submitted via the eLearn@USM by one of the group members:

1. An assignment report (not more than 20 pages) must be prepared using any word editor (e.g. Microsoft Word, Open Office and etc). The report must explain the Simulated Annealing algorithm and the experiments performed to obtain the best results. All the important aspects must be explained such as initialization, a state neighbourhood, acceptance probabilities, the annealing schedule, parameter tuning, and penalty function design. Sample output screen shots and instructions on how to run your program must be included in the assignment script. Also, a **cover page** which contains your details is expected to be included to your assignment script.
2. Program source files.
3. The zip/rar package must be named according to the following notation: **CPT244_*MatricNo1_MatricNo2*_P01**. For example, two students with matric numbers 111222 and 111333 are forming a group; the zip/rar package is named as **CPT244_*111222_111333*_P01**.

Assignment Evaluation: This assignment will be graded (A to F scale) and the total will be scaled to 10% of your overall grade.

Reference: Kindly state any source of reference in your assignment script should you refer to various sources to complete this assignment.

An oral examination will be taking place. All the groups must book a session so that the project is explained. Further details will be announced.

IMPORTANT: Students who copied or plagiarized other's work or let their work be copied or plagiarized will be given an F grade. The student may be barred from sitting for final exam and reported to the university's disciplinary board.

Sample Source code for Genetic Algorithm:

1. Simple GA to solve card problems (C#)
http://www.codeproject.com/Articles/16286/AI-Simple-Genetic-Algorithm-GA-to-solve-a-card-pro
Source code: https://www.dropbox.com/sh/jt46adby7qn6flu/AADU8ZnH8PqgNIOSYPc2XXSLa

2. Simple Binary GA (Java)
http://kunuk.wordpress.com/2010/09/27/genetic-algorithm-example-with-java/
Source code: given in the website page

~~END OF PROJECT 01~~