

Alunos: Elias Fank; João Gehlen; Ricardo Zanuzzo

Representação dos gens dos indivíduos:

Um indivíduo é formado pelo conjunto de todas as fases de um semestre de Ciência da Computação. Cada uma dessas fases possui um conjunto de horários possíveis que formam as gens do indivíduo.

Como foi feito o cruzamento entre indivíduos:

No nosso projeto o cruzamento é feito da seguinte forma: O sistema pega aleatoriamente dois indivíduos da população. Depois de cada um desses dois indivíduos selecionados, o sistema pega uma fase(semestre) de cada indivíduo, ou seja, o semestre 1 do indivíduo 1, o semestre 2 do indivíduo 2, o semestre 3 do indivíduo 1, o semestre 4 do indivíduo 2... e assim consecutivamente até ter todas as fases para gerar um novo indivíduo. Resumindo, faz um “merge” entre dois indivíduos para gerar um novo. Depois de criado o novo indivíduo, o sistema remove da população aquele que tiver a pior fitness. E então adiciona o novo indivíduo criado.

Abaixo segue um exemplo para auxiliar o entendimento:

Indivíduo 1 = [S1i, S2i, S3i, S4i, S5i, S6i, S7i, S8i, S9i], onde ‘i’ significa indivíduo 1.

Indivíduo 2 = [S1j, S2j, S3j, S4j, S5j, S6j, S7j, S8j, S9j], onde ‘j’ significa indivíduo 2.

novo Indivíduo = [S1i, S2j, S3i, S4j, S5i, S6j, S7i, S8j, S9i]

Depois de criado esse novo Indivíduo, o sistema coloca ele na população e remove da população aquele indivíduo que tiver pior a fitness.

Como foi feita a mutação dos indivíduos:

Ao fazer o cruzamento dos indivíduos, e um novo indivíduo ser gerado, ele pode com uma certa probabilidade (definida no algoritmo como *probMutation*), realizar ou não uma mutação. Caso aconteça, a mutação ocorre da seguinte forma: os gens

(horários) atuais do indivíduo, juntamente com os horários disponíveis que compõem aquele semestre, são colocados em uma lista. Para cada gem do indivíduo, também existe uma probabilidade (nomeada de *mutationRate*) de ocorrer uma mutação. Com isso, se a mutação ocorrer, o sistema escolhe aleatoriamente um elemento da lista, caso o elemento escolhido for um horário livre, ele é substituído pelo gem atual. Caso o elemento escolhido já estiver presente no gem, então é realizada uma troca de posição entre eles, ou seja, os gens são permutados entre si.

Para exemplificar a mutação, suponha que o novo indivíduo gerado é composto dos seguintes gens (H1, H2, H5, H6), sendo H os horários do semestre, e a lista com todos os horários é [H0, H1, H2, H3, H4, H5, H6]. Para o horário H1, será feita a mutação e o sistema escolhe da lista o horário H3, supondo que H3 é um horário disponível, então será feita a substituição do gem H1 pelo H3. Os gem H2 e H5 não realizaram mutação. O próximo gem H6 sofre mutação, e é escolhido o horário H2 da lista, como H2 já faz parte do indivíduo, eles serão permutados. Com isso, ao fim da mutação o indivíduo ficará assim (H3,H6,H5,H2).

Após ser gerado o novo indivíduo no cruzamento, e com uma certa probabilidade realizar ou não uma mutação, será feito o cálculo da fitness sobre ele.

Como foi feita a função fitness:

Para todos os novos indivíduos gerados no algoritmo, será realizado o cálculo da fitness sobre ele. Quanto menor a fitness melhor é o indivíduo, pois o cálculo da fitness é feito seguinte forma :

- Caso aconteça uma quebra de restrição o indivíduo é penalizado com **1000** a mais na fitness.
- Caso aconteça uma quebra de preferência do tipo:
 1. "*De preferência os professores não devem ministrar 4 períodos consecutivos no mesmo turno*" - O indivíduo é penalizado com **10** a mais na sua fitness.

2. *"De preferência os professores não devem ministras aulas no período matinal e noturno no mesmo dia"* - O indivíduo é penalizado com **2** a mais na fitness.
3. *"De preferência os professores não devem ministrar aulas nos períodos que eles solicitaram não ministrar aulas."* - O indivíduo é penalizado com **1** a mais na fitness.

Parâmetros usados na execução do algoritmo:

Os parâmetros foram ajustados até que uma fitness apropriada fosse encontrada.

Os parâmetros são:

- `int quantIndividuos = 1000;`
- `double probMutation = 0.15;`
- `double mutationRate = 0.03;`
- `int numCruzamentosporGeracao = 10;`
- `int numCruzamentos = 150;`
- `int maxGenerations = 100;`
- `int condParada = 10;`

Para compilar:

Por SH:

```
$ ./compilar-executar.sh
```

Por Makefile:

```
$ make all
```

```
$ make run
```

Resultado Final

O algoritmo mostra a evolução das gerações, mostrando a melhoria da fitness...

- Abaixo segue uma Tabela mostrando o resultado do algoritmo.
- Legenda:
 - **Laranja**, sofreu quebra da preferência nº 1, somado +10 na fitness;
 - **Amarelo**, sofreu quebra da preferência nº 3, somado +1 na fitness.

| V1 | Seg | Ter | Qua | Qui | Sex |
|-------|-----------------|----------------|----------------|-------------|----------------|
| Manhã | | | | | |
| | | | | | |
| Tarde | CD - Padilha | CD - Jacson | Alg - Priscila | CD - Jacson | Alg - Priscila |
| | Alg - Guilherme | Alg - Priscila | Il - Raquel | Il - Raquel | CD - Jacson |
| Noite | | | | | |
| | | | | | |

| M3 | Seg | Ter | Qua | Qui | Sex |
|-------|-------------|-----|-------------|-----------------|-----|
| Manhã | OC - Jacson | | ED2 - Lais | ED2 - Lais | |
| | Prg1 - Lais | | OC - Jacson | Prg1 - Priscila | |
| Tarde | | | | | |
| | | | | | |
| Noite | | | | | |
| | | | | | |

| M5 | Seg | Ter | Qua | Qui | Sex |
|-------|---------------|-------------|--------------|----------------|--------------|
| Manhã | | BD2 - Denio | ES1 - Raquel | SO - Marco | ES1 - Raquel |
| | | | BD2 - Denio | IPC - Graziela | |
| Tarde | SO - Marco | | | LFA - Braulio | |
| | LFA - Braulio | | | IPC - Graziela | |
| Noite | | | | | |
| | | | | | |

| M7 | Seg | Ter | Qua | Qui | Sex |
|-------|---------------|------------------|------------------|-----|-----------------|
| Manhã | TCC1 - Marco | Opt1 - Guilherme | IA - Bins | | Opt2 - Andressa |
| | Dist - Emilio | Opt2 - Padilha | Opt1 - Guilherme | | TCC1 - Marco |
| Tarde | IA - Bins | | | | |
| | Dist - Emilio | | | | |
| Noite | | | | | |
| | | | | | |

| N2 | Seg | Ter | Qua | Qui | Sex |
|-------|-----|--------------|-----|-----|-------------|
| Manhã | | | | | |
| | | | | | |
| Tarde | | | | | |
| | | | | | |
| Noite | | SD - Emilio | | | ED1 - Lais |
| | | ED1 - Jacson | | | SD - Emilio |

| N4 | Seg | Ter | Qua | Qui | Sex |
|-------|-----|-----|------------------|--------------|--------------|
| Manhã | | | | | |
| | | | | | |
| Tarde | | | | | |
| | | | | | |
| Noite | | | OC - Padilha | OC - Padilha | BD1 - Denio |
| | | | Prog2 - Priscila | BD1 - Denio | Prog2 - Lais |

| N6 | Seg | Ter | Qua | Qui | Sex |
|-------|-----|-----|----------------|-----|----------------|
| Manhã | | | | | |
| | | | | | |
| Tarde | | | | | |
| | | | | | |
| Noite | | | ES2 - Graziela | | ES2 - Graziela |
| | | | LFA - Braulio | | LFA - Braulio |

| N8 | Seg | Ter | Qua | Qui | | Sex |
|-------|--------------|--------------|-----|-------|---------|-----|
| Manhã | | | | | | |
| | | | | | | |
| Tarde | | | | | | |
| | | | | | | |
| Noite | Opt1 - Pavan | Opt1 - Pavan | | Red - | - Pavan | |
| | PGP - Raquel | PGP - Raquel | | Red - | - Pavan | |

| N10 | Seg | Ter | Qua | Qui | Sex |
|-------|-------------|--------------|--------------|------------------|------------------|
| Manhã | | | | | |
| | | | | | |
| Tarde | | | | | |
| | | | | | |
| Noite | | TCC2 - Marco | | Opt4 - Guilherme | |
| | Opt3 - Bins | | TCC2 - Marco | Opt3 - Bins | Opt4 - Guilherme |

- Fitness = 17, uma quebra preferência nº 1 (+10) e 7 quebras preferência nº 3 (+7).

Exemplo de saída (Resultado Final):

Geração: 1 Fitness: 24
Geração: 2 Fitness: 24
Geração: 3 Fitness: 24
Geração: 4 Fitness: 24
Geração: 5 Fitness: 24
Geração: 6 Fitness: 24
Geração: 7 Fitness: 24
Geração: 8 Fitness: 24
Geração: 9 Fitness: 23
Geração: 10 Fitness: 23
Geração: 11 Fitness: 23
Geração: 12 Fitness: 23
Geração: 13 Fitness: 23
Geração: 14 Fitness: 23
Geração: 15 Fitness: 22
Geração: 16 Fitness: 22
Geração: 17 Fitness: 22
Geração: 18 Fitness: 22
Geração: 19 Fitness: 21
Geração: 20 Fitness: 21
Geração: 21 Fitness: 19
Geração: 22 Fitness: 19
Geração: 23 Fitness: 19
Geração: 24 Fitness: 19
Geração: 25 Fitness: 19
Geração: 26 Fitness: 19
Geração: 27 Fitness: 19
Geração: 28 Fitness: 19
Geração: 29 Fitness: 19
Geração: 30 Fitness: 17
Geração: 31 Fitness: 17
Geração: 32 Fitness: 17
Geração: 33 Fitness: 17
Geração: 34 Fitness: 17
Geração: 35 Fitness: 17
Geração: 36 Fitness: 17
Geração: 37 Fitness: 17
Geração: 38 Fitness: 17
Geração: 39 Fitness: 17
Geração: 40 Fitness: 17
Geração: 41 Fitness: 17
Geração: 42 Fitness: 17

#RESULTADO#####

Fase: V1

11: Alg - Guilherme

12: CD - Jacson

13: Alg - Priscila

14: Alg - Priscila

15: Il - Raquel

16: CD - Jacson

17: Il - Raquel

18: Alg - Priscila

19: CD - Jacson

10: CD - Padilha

Fase: M3

00: OC - Jacson

01: Prg1 - Lais

04: ED2 - Lais

05: OC - Jacson

06: ED2 - Lais

07: Prg1 - Priscila

Fase: M5

11: LFA - Braulio

02: BD2 - Denio

04: ES1 - Raquel

05: BD2 - Denio

16: LFA - Braulio

17: IPC - Graziela

06: SO - Marco

07: IPC - Graziela

08: ES1 - Raquel

10: SO - Marco

Fase: M7

11: Dist - Emilio

00: TCC1 - Marco

01: Dist - Emilio

02: Opt1 - Guilherme

03: Opt2 - Padilha

04: IA - Bins

05: Opt1 - Guilherme

08: Opt2 - Andressa

09: TCC1 - Marco

10: IA - Bins

Fase: N2

22: SD - Emilio

23: ED1 - Jacson

28: ED1 - Lais

29: SD - Emilio

Fase: N4

24: OC - Padilha
25: Prog2 - Priscila
26: OC - Padilha
27: BD1 - Denio
28: BD1 - Denio
29: Prog2 - Lais

Fase: N6
24: ES2 - Graziela
25: LFA - Braulio
28: ES2 - Graziela
29: LFA - Braulio

Fase: N8
22: Opt1 - Pavan
23: PGP - Raquel
26: Red - Pavan
27: Red - Pavan
20: Opt1 - Pavan
21: PGP - Raquel

Fase: N10
22: TCC2 - Marco
25: TCC2 - Marco
26: Opt4 - Guilherme
27: Opt3 - Bins
29: Opt4 - Guilherme
21: Opt3 - Bins

#####