

- (4 pts) Given a target string, the goal is to produce it starting from a random string of the same length. Use **Genetic Algorithm** to solve this problem. The algorithm terminates when the number of iterations is larger than **5000**, or the target string is generated. The details of implementation are as follows.

- Characters A-Z, a-z, 0-9 and other special symbols are considered as genes. The target string is "Hello, World! 19/11/09".
- A string generated by these characters is considered as chromosome.
- Fitness score is the number of characters which are equal to characters in target string at a particular index. For example, the fitness score of "ke:lF,)5X#iM#&\$n11g0A" is 6.

H	e	l	l	o	,	W	o	r	l	d	!	-	9	/	1	1	/	1	0	
k	e	:	l	f	,)	5	X	#	i	M	#	&	\$	n	1	1	g	0	a

Requirement: Print the individual of the highest fitness score in each generation. For example,

Generation: 1 String: H6f7Vu1Fol1eZB#g73SK#z

⋮

Generation: 54 String: ke:lF,)5X#iM# &\$n11g0A

⋮

Generation: 396 String: Hello, Noeld! g9/11/09

⋮

```

1 // Valid Genes
2 const string GENES = "ABCDEFGHJKLMNOPQRSTUVWXYZ\"
3 "abcdefghijklmnopqrstuvwxyz 1234567890 , . - ; : _ ! \" # % & / ( ) = ? @ $ { [ ] } " ;
4
5 // Target string
6 const string TARGET = "Hello , World! 19/11/09" ;

```

- (4 pts) Find x^* to minimize $x \cdot \sin(x)$, $x \in [-1, 15]$ ($\sin(\pi) = 0$). Use **Genetic Algorithm**. Chromosome is a 16-bit binary code. The algorithm terminates when the number of iterations is larger than **5000**, or the average fitness score \approx the best fitness score.

Requirements:

- Design the implementation details by yourself.

- $x^* = \arg \min_x x \cdot \sin x \approx 11.0857$. It will be accepted as long as your final result falls into $(11.08, 11.10)$.
 - Print the individual of the best fitness score in each generation.
3. (2 pts) Describe your implementation of Question [2](#), including the end criterion, crossover operator, selection methods and so on.

Important Notes:

- **Reference:** The class note of Lecture 15.
- Remember to submit your **makefile**!
- Submit question [3](#) in a *pdf* file.
- Due: **2019/11/17 11:59pm**