NAT 2011: Assignment 1

This is an exercise worth 10% of the course mark. It requires you to write a program and to use it to carry out a number of investigations. You are to write up your investigations in the form of a conference paper with a maximum length of 6 pages (12 pt., one- or two-column layout). The page limit is strict and marking will be done based on these 6 pages. You can submit more detailed descriptions in form of appendices. In addition to the report, submit also the code of your program(s). Both should be done electronically using the submit program: 'submit nat 1 <filename>', see the man page of the submit program for more information. Please submit your report as a pdf file and your code separately as a plain text file or an archive of text files.

You should submit your assignment by 4pm of Thursday, 27. October, 2011.

The exercise is about the **prisoner's dilemma** (PD). This problem has been studied by many researchers, some of them using genetic algorithms. Feel free to search through the literature, but make sure to cite all the papers that you found useful for this project. The list of references (as a kind of appendix) will not count towards the 6 page limit, but you should place a pointer (cite or endnote) in the main text where you describe in what way the referred literature was useful.

The task is to evolve strategies for playing the following game. Each player chooses either to "cooperate" or to "defect" and receives a reward according to the following table:

Payoff matrix		Player A	
		Cooperate	Defect
Player B	Cooperate	*C*,*C*	*T*,*S*
	Defect	*S*,*T*	*D*,*D*

Table 1: The Pay-off matrix for the prisoner's dilemma.

The first of the entries is the reward ('reduction of sentence') player A receives, second entry is for player B. The rewards are usually chosen subject to the conditions $S < D < C < T$ and $T + S < 2C$, e.g. S=0, D=1, C=3, T=5. The maximal overall reward seems to be achievable if both players cooperate, but anticipating cooperation of the other one a player may be tempted to defect in order to increase its own fitness. Please check wikipedia and other sources for explanations of the game.

The iterated PD (IPD) consists of a series of games, where often a player knows the previous decisions of the opponent and can use this information to form its strategy. We will study the IPD for a population of players which choose their partners randomly or according to certain rules.

The problem (except perhaps for the first subtask) is not the most typical application of a GA. Many applications of GAs consist in searching for near-optimal solutions, while here a global optimum might never be reached. The IPD can be considered as a problem that highlights some of the features of GAs especially in relation to natural (biological and memetic) evolution. Nevertheless, the fitness function for the problem is obvious such that a GA are readily applicable.

In the report on your work on this assignment write separate sections answering the following questions and include graphical representations of your numerical results.

1. (Introduction) What problem is approached by what means? Why it might be interesting to study the problem? Keep this part very brief. Do not extensively repeat information from the lectures. (marks: 5/100)
2. How can a strategy be represented if it is supposed to use information about previous decisions[1] of the opponent? (5/100)

1 We assume that full information about the past ("Timeline") is available for all agents (indviduals).

3. In order to test your algorithm start with a simple “all ones” version, i.e. an agent should cooperate here (choose e.g. *S*=0, *D*=0, *C*=1, *T=0)* irrespectively what the opponent did. Use here a fixed history of e.g. five previous decisions. (10/100)

Back to the IPD, note that (especially for long histories) the algorithm may need many steps before it behaves independently from the (random) initial conditions. When choosing parameters, run time should be one criterion. Yet, make sure to use low mutation rates and medium population sizes (why?). It would be good to set up the algorithm such that different string lengths (or history lengths) are possible, but in some of the problems you may use as well constant lengths. (Before you continue, do not forget to set *S*, *D*, *C* and *T* back to values according to the above conditions.)

4. Strategies may use different numbers of past decisions. How would you set up selection (fitness), mutation and crossover in this case? (10/100)
5. What behaviours can you observe when running the algorithm sufficiently long? Can you characterise these behaviours (e.g. mean and variance of the fitness, by schemata that are dominant in the population, by a clustering of the individuals or in any other way)? Try to explain your observations. (20/100)
6. The observable behaviour of the algorithm depends on the parameters and the set-up of the algorithm (populations size, mutation, crossover rate, selection mechanism, history length, fitness function, encoding etc.). Choose three variations of the algorithm, present graphically and explain the results of your simulation of the effects. (20/100)
7. The problem is particularly interesting if a neighbourhood is used in selection and reproduction. The population could be distributed on a two-dimensional grid (torus) (e.g. such that an individual interacts only with their four or eight neighbours). How does the behaviour differ from the all-to-all interaction? Can you find a (near) stationary state of the algorithm where not all individuals use a similar strategy? Can you find spatio-temporal patterns? (20/100)
8. (Discussion) Discuss here question such as: Why did the algorithm not perform as expected? Why did the algorithm perform as is did? What is the role of ‘fitness’ in the IPD? How (or how not) to interpret the results? (etc.) (5/100)
9. (Conclusion) What are the main results in words? (5/100)

Please state clearly (e.g. simply by following the numbering of the tasks) which part of your report answers what question. Try to write your report such that it is understandable even without repeating the questions (assuming the reader is familiar with GAs in general). You should describe the experimental design, the results, and discuss the results. Use figures where appropriate. It will be helpful if you comment your code, but there will be no marks for good code (you may loose marks if neither the text nor the code is clear).

If you need to choose values of parameters of the GA, you should explain and justify how you came to this choice, given that you (or rather, the individuals in the population) are aiming at good performance. You also need to decide how you are going to present results that such that you present a clear overall picture of the behaviour of the GA in the present problem. Check Mitchell's book, Chapter 5, for issues to consider when designing a GA.

References (Obviously, more references have been used, please be more complete in your report.)
David Fogel. 1993. Evolving behaviors in the iterated prisoner's dilemma. *Evol. Comput.* 1, 77-97.

NB: The following criteria will be used in evaluating your assignment: Is the design of the algorithm sound? How clearly is your design and implementation described? How sound are the experiments? Are the results statistically significant? (Note that averaging is not always useful for the IPD. Why?) To what extent did your experiments give the GA “a good workout” (i.e. do you understand the effect of the parameters and the design on the results)? Does the discussion of your results shed light on how the GA works as you vary the parameters and the difficulty of the problem?