UNIVERSITY OF JOHANNESBURG

FACULTY OF SCIENCE

ACADEMY OF COMPUTER SCIENCE AND SOFTWARE ENGINEERING

MODULE	**IT00107** OPTIMISATION
CAMPUS	**APK**
EXAMINATION	**JANUARY, 2017**

DATE 10 JANUARY 2017	**SESSION** 8:30 – 10:30
ASSESSOR(S)	DR. GJ VAN NIEKERK
INTERNAL MODERATOR	NONE
EXTERNAL MODERATOR	DR. WJC VAN STADEN (UNISA)
DURATION 2 HOURS	**MARKS** 100

NUMBER OF PAGES: 6 (cover page included)

INSTRUCTIONS

ANSWER ALL THE QUESTIONS
WRITE NEATLY AND LEGIBLY
NUMBER IN SEQUENCE

REQUIREMENTS

NONE

Section A: Evolutionary Algorithms

1. The representation (or chromosome) of a solution is critically important. Describe clearly how a solution for each of the following problems can be represented in an actual implementation.
   (a) (2 marks) Evolving a line of text to match some predefined piece of text of a predetermined length.
   (b) (2 marks) Evolving a mathematical equation to find a sequence of numbers that adhere to some property (for example perfect squares or triangle numbers, etc.)
   (c) (4 marks) Evolving an image by randomly placing a set number of semi-transparent multi-coloured circles on a square.

2. (4 marks) For Steady-state GAs (SSGA) the kill-tournament replacement strategy is often used. Explain how the replace-worst and replace-random strategies are special cases of the kill-tournament and why a kill-tournament is a better than both of these.

3. (10 marks) Design a Cooperative Co-Evolutionary Differential Evolution Algorithm for two populations (i.e. combine Co-evolution and Differential evolution in the same algorithm). Give pseudo-code and explain each part of your algorithm.

4. (2 marks) Briefly explain why interactive GAs requires human intervention. Give an example of a problem that will require this kind of interaction.

5. (15 marks) The Appolo 13 mission has suffered catastrophic failure and the pilots are counting on you to save their lives. The module needs to survive re-entry, but only has limited power remaining (a total of **513** watt-hours) due to damage to the system. Several systems needs to be switch on without exceeding the maximum power available. One of the electrical engineers has supplied a table showing the importance of each system. A small subset of the table is shown below.

Component	**Power (Watt-hours)**	**Importance**
CO2 scrubber	21 Wh	Medium
Guidence system main computer	67 Wh	Critical
Guidence system supplementary computer	13 Wh	Medium
Hatch indicator light	2 Wh	Low
Heating	240 Wh	Medium
Main computer	65 Wh	High
Thermostat	12 Wh	Medium
...	...	...

   There are a total of **212** such components, each with power requirement and a priority (Low, Medium, High and Critical). To maximise the chances of survival, the subset of components to switch on must be carefully chosen. If the power exceeds 513 Wh, the power system will overload and the module will burn out on re-entry.

Design a Genetic Algorithm that will be able to find the combination of systems that needs to be switched on to maximise the chances of survival. Clearly elaborate on the **representation scheme**, the **fitness function** and any specifics regarding the reproduction and selection operators.

6. (15 marks) The Eaters' World is divided into little squares. Each square can hold an eater, a plant, or it can be blank. Plants tend to grow in some predefined way. A square cannot be occupied by two things at once (see figure 3).

   An eater can only "see" the single square just in front of it. (The front is the pointy end of the T-shaped Eater.) It can see one of four different things: another eater, a plant, a blank space, or a wall (i.e. the sides of the screen). In addition to this external information, the eater has an internal memory that contains a number between 0 and 15. This number is called the "state" of the eater. At each time step, the eater can perform **one** of the actions:

   - Move forward one square
   - Move backwards one square
   - Turn in place 90 degrees to the left
   - Turn in place 90 degrees to the right.

   Apart from these four actions, it can also change its state by changing the number in its memory. If it tries to move into a wall or onto a square that already contains another eater, it will not be allowed to move; however, it can still change its state. If it moves onto a square containing a plant, it automatically "eats" the plant and scores a point.

   An eater always bases its decision on two things: its **current state** and the **item that it sees in front of it**. For example, a hypothetical eater may see another eater in front of it and have a current state of 7. Given the other eater and the internal state of 7, it decides to move backwards one square.

   Using any Evolutionary Algorithm of your choice, give a detailed discussion on how you will evolve the best eaters. Highlight all aspects of your algorithm, namely the **paradigm** and the reasons for choosing it; **the representation of the chromosome**; the **fitness function** and the **reproduction operators**. Discuss how you would wire all of this together.

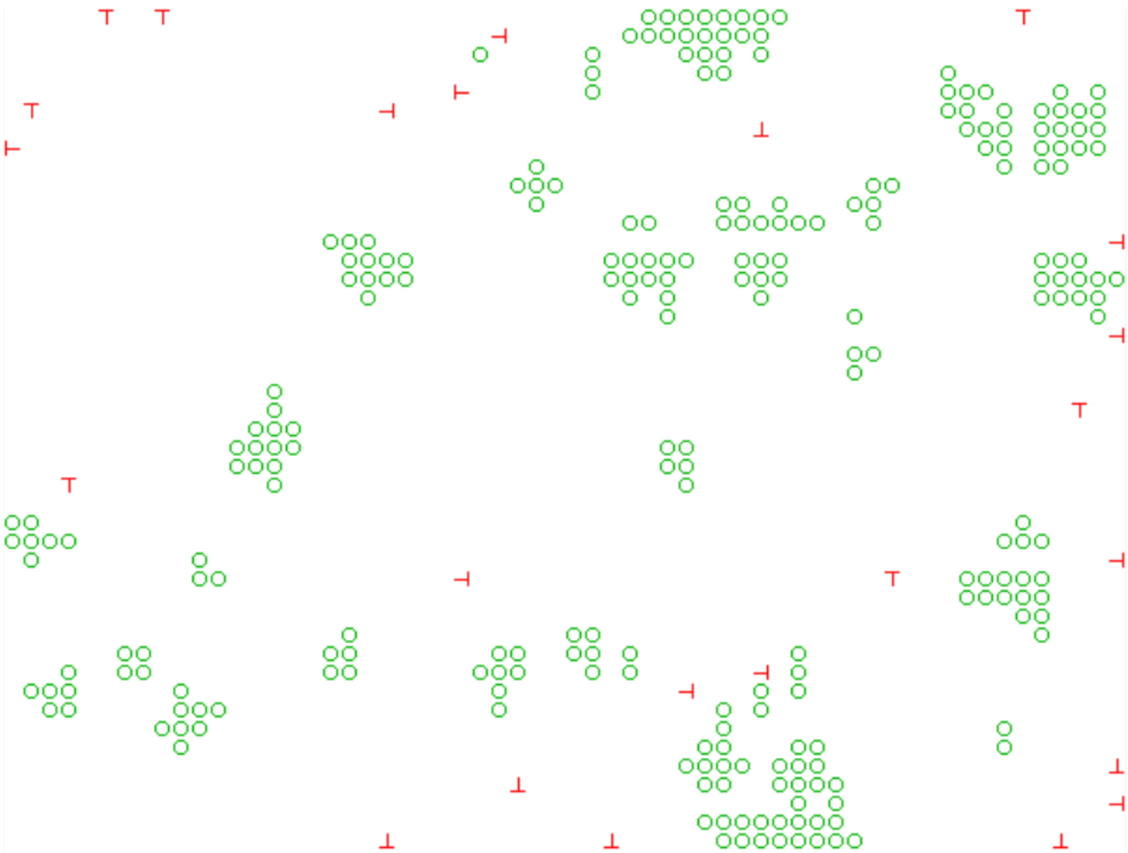

Figure 1: Eater's world

Section B: Particle Swarm Optimisation

1. Using indices to form local neighbourhoods, information (specifically the position of the global best) tends move slower through the "lbest" algorithm as oppose to the "gbest" algorithm.
   (a) (2 marks) Briefly explain why this happens.
   (b) (2 marks) Is the slower movement an advantage or a disadvantage (or both)? Motivate your answer clearly.
2. (4 marks) Explain the significance of the exploration/exploitation trade-off in the context of PSOs and discuss the problems that arise if this trade-off is not in balance.
3. (6 marks) Briefly discuss each of the three components that make up the velocity, $v_{ij}(t)$ of each particle in the standard 'gbest' PSO and the effect that each has on the velocity of the particle.
4. (6 marks) Briefly explain how the Cooperative Split PSO (CPSO) algorithm works. Place special emphasis on the unique aspects of the algorithm such as the representation and the fitness function. Also elaborate on the advantages of the approach.

Section C: Ant Algorithms

1. (3 marks) Briefly explain the concept of 'emergent behaviour' in terms of ant algorithms and give an example to illustrate your point.
2. (2 marks) Discuss the importance of evaporation in the pheromone trail depositing equation of ant-based optimisation.
3. *Dorigo et al.* proposed three different pheromone update rules, namely Ant-cycle, Ant-density and Ant-quantity Ant System.
   (a) (3 marks) Briefly explain the differences between three update rules.
   (b) (3 marks) Considering the classic Travelling Salesperson problem (all cities viable from every city), which of these three update techniques will yield the best result. Motivate your answer clearly.
4. (15 marks) Mobile networks are separated into cells. Each cell has a set of frequencies that it uses for communication. For example, a small cell could have $(3, 8, 12)$ as its assigned frequencies or $(4, 12, 18, 22, 34, 50, 53, 60, 72)$ if it is a bigger cell. The frequencies are simply identified with a number.

It is often useful to group cells in a mobile-phone network based on the their assigned frequencies. For example, two cells with frequency sets $(4, 8, 12)$ and $(5, 8, 12, 18)$ may be grouped together since both cells have 8 and 12 in common, 4 and 5 are close together, while the larger cell only has one additional frequency, namely 18.

Assume that you have 30,000 cells in the network and that each cell has 3 to 12 assigned frequencies from a pool of 100 (numbered 1 to 100).

Discuss a clustering algorithm based on the behaviour of ants that can be used to separate the cells into an arbitrary number of groups. Explain in detail how the data points needs to be modelled and how dissimilarity between these data points can be established.