****

**Student Presentations – EPSRC Vacation Scholars**

|  |  |
| --- | --- |
| Name | Emilia Vulpe |
| Student ID | 1106723 |
| School | **School of Computing Science** |
| Title of Presentation | Relaxed Garbage Collection |
| Project Supervisor(s) | Jeremy Singer |
| Selected Format for Presentation (poster/oral) | Oral |

**Abstract (up to 500 words)**

The abstract/presentation/poster should address the following questions where relevant as well as providing a more general overview of your presentation:

1. What were the challenge/s the research addressed?
2. What was the purpose of your research?
3. Why was your research important and what is the potential impact?
4. Research questions/hypotheses
5. Any results

|  |
| --- |
| 1. Relaxed garbage collection is an automatic way for memory management. The way it works is when a program is run, every now and then it freezes and garbage collector looks through objects in memory that are dead and deallocates them to free more memory for new objects that are needed for the program. The problem with this approach is that the process of looking for dead objects in memory is slow and may increase significantly the running time of the program. What we introduce is a new policy for deletion of objects by implementing different heuristics for prediction of when objects won’t be used by the program before they die and remove them from memory rather than perform the slow check for dead objects every time garbage collection takes place. 2. The way we aim to remove objects from memory may lead to errors caused by removing objects that may be needed by the program at a later stage so the purpose of the research was to identify how to minimize the percentage objects causing such errors leading to an optimal performance of the program. In other words, we wanted to avoid the slow process of traditional garbage collection where the collector freezes the program until it finishes its job and predict which objects are not needed any more and deallocate them before they actually die thus not freeze the program too often. 3. The research was important because it can lead to a great improvement of the way garbage collection works and increase the performance of programs without any additional cost. 4. The main hypothesis was that a big percentage of objects live for a short amount of time and after this time they are not accessed any more so they can be removed safely from memory. Another hypothesis was that there is a correlation between the objects’ sizes and the time they live for. Also we supposed that the memory size has a great impact on the percentage of errors that can be identified after deleting objects from memory. 5. The results prove that the described above hypotheses are correct. Having a big enough memory, around 175 or 200 megabytes, will allow objects to live long enough and if we remove the first objects allocated which coincides with the least recently used ones we are not introducing errors so for now this is the optimal solution but needs to be investigated further. |