

# NUI Galway

## Programming and Tools for AI (CT5132/CT5148)

James McDermott

December 4, 2020

### Assignment 3: Hand-coding solutions for the Abstraction and Reasoning Corpus

**Weighting** 25% of the module

**Deadline** Midnight Sunday 13th December, end of Week 11 (Sunday chosen for benefit of students in full-time employment).

**Background** The “Abstraction and Reasoning Corpus” (ARC) of Chollet is an interesting new benchmark for AI research.

- See some example tasks at the end of this document.
- See <https://github.com/jmmcd/ARC> for the repository we will use. (The original repository is at <https://github.com/fchollet/ARC> but we will not use that here.)
- To access the data and helper code, you should **fork** the GitHub repository to your own GitHub account and then **clone** it to your hard disk. Then you can open (double-click) `apps/testing_interface.html` to see some tasks and try to solve them by hand.

ARC consists of many tasks. Tasks involve small grids of coloured squares (encoded as JSON objects with colours encoded as integers 0-9). For each task, there are several “training” input-output grid pairs which demonstrate, given an input grid, the correct output grid. There are then one or more further “test” grid pairs.

Solving most tasks is quite easy for a human. The real goal of ARC is to create an AI system which is able to solve any such task by learning on the demonstration pairs (perhaps by generating a task-specific program for each task). This is difficult, beyond the current state of the art. Optionally, see Chollet *The Measure of Intelligence* <https://arxiv.org/abs/1911.01547> for further background and motivation.

**Requirements** For this assignment, students will choose any 3 or more tasks from the `data/training` directory (we will not use the `data/evaluation` directory at all), and write a *hand-coded* solution for each task.

Some tasks are very easy and some more difficult. Some marks will be awarded for the degree of difficulty of tasks chosen.

This solution will consist of a `solve` function for each chosen task. It must be named `solve_<ID>`, where `<ID>` is the task ID, e.g. `solve_017c7c7b`. The only argument must be `x`, a single grid, encoded as a 2D Numpy integer array. It should return a single grid in the same format.

To avoid trivial solutions: the `solve_*` functions must not work by looking up the correct solutions, or contain any portion of the correct solutions as hard-coded data.

The `solve_*` functions may use any programming techniques studied in class or elsewhere. They may or may not require any machine learning or AI methods. Your code may **import** from any library in the Python standard library, or any library installable via Anaconda or Pip, but may not use or import any previously-written code or library specific to ARC, including previous submissions to this assignment.

These functions must be written in the `src/manual_solve.py` file which already contains helper functions for loading ARC JSON data and testing.

To access the data and helper code, you must **fork** the GitHub repository to your own GitHub account and then **clone** it to your hard disk. As you work, you must **commit** your work in progress as appropriate and **push** to GitHub. Also, make some small, appropriate modifications to the `README.md` file to reflect the purpose of your repository in contrast to the original.

Your `manual_solve.py` should contain comments or docstrings including:

- Student name(s) and ID number(s).
- A link to the URL of your GitHub repository.
- For each chosen task:
  - A verbal description of the required transformation (approx. 1-3 sentences).
  - Normal comments describing how `solve` carries out the transformation (approx. 3-5 sentences).
  - A statement of which training and test grids are solved correctly.
- A short summary/reflection, commenting on the Python features and libraries you used in the `solve_*` functions and on any commonalities or differences among them (approx. 2-3 short paragraphs). This should be about the `solve_*` functions, not about use of JSON or testing infrastructure etc.

**Grading** Marks will be awarded for:

- Solving tasks and degree of difficulty of chosen tasks (40%) (an incomplete attempt at three difficult tasks will receive more marks than a perfect solution for three very easy tasks – see Appendix for examples of difficulty);
- Correspondence between the task, the verbal description of the transformation, the description of how `solve_*` works, and the code itself (20%).
- Understanding of commonalities in the summary (10%).
- Good quality code including readability (10%).
- Use of git/GitHub including README (20%).

Some bonus marks may also be awarded for demonstrating understanding (in comments/docstrings or README) of issues raised in the Chollet paper or elsewhere in the AI research literature and their relevance to your hand-coded solutions. These will be “bonus marks” in the sense that these issues are outside the scope of the module and full marks can be achieved without any bonus marks.

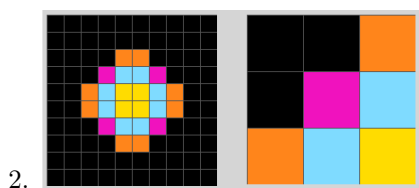
**Submission** Submit only the `manual_solve.py` file which must include a link to your GitHub repository. Do not submit data files, html, or css, or a zip. Submit by going to Blackboard - Assessment - **Assignment 3**.

**Groups** Groups of 1 or 2 students are allowed. However, your group must not include any student you worked with in any Assignment in this module or any other. If using a group of 2 students, the GitHub repo must demonstrate group collaboration, i.e. multiple commits from each group member. In a group, only one student need submit, but if desired (to silence Blackboard alerts), both may submit identical submissions.

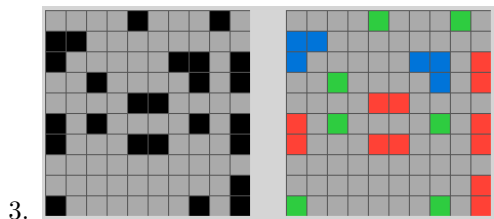
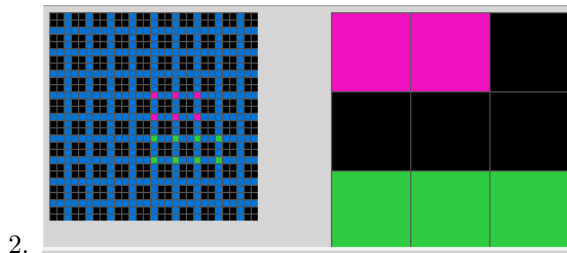
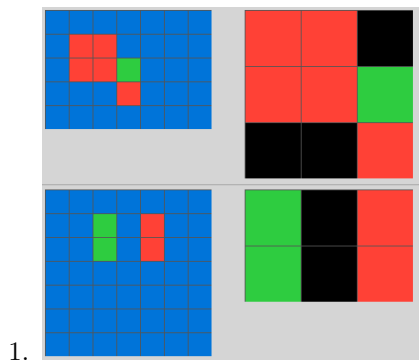
**Plagiarism** Students are reminded of the University’s policy on plagiarism. Students may discuss the assignment with other students but must not look at other students’ work, or allow others to look at theirs. Any online sources used must be cited with URL and date of access in a comment. **Any use of previous solutions of the tasks is disallowed.** Materials from CT5132/CT5148 need not be cited. By making a submission, you declare that you have abided by these conditions. Suspected infringements will be investigated and may be referred to NUI Galway authorities.

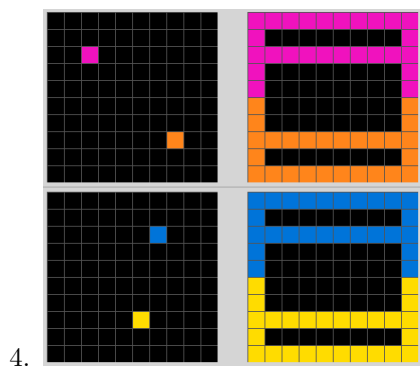
## Problem difficulty

### Easy examples

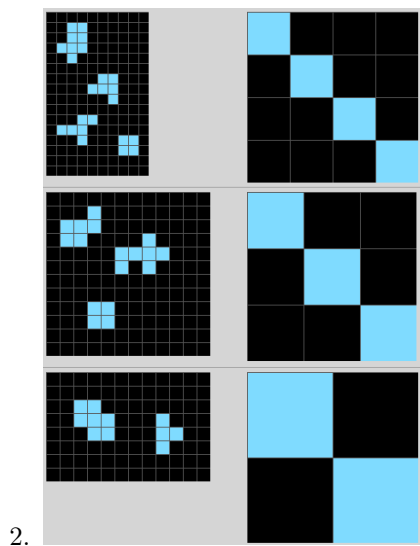
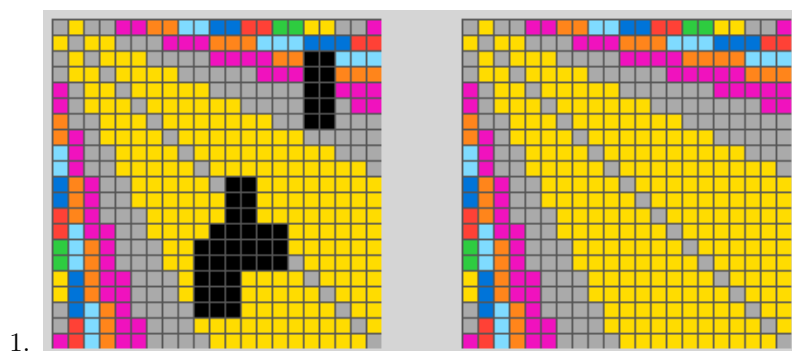


### Medium examples

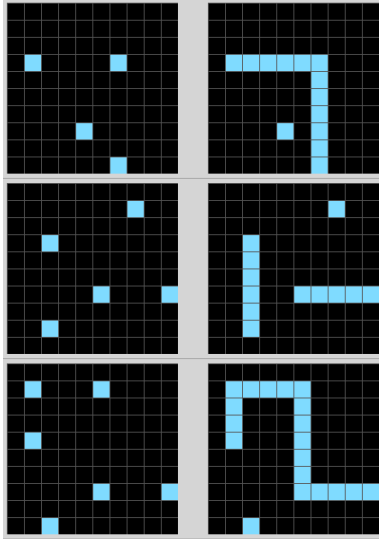




# Medium-to-difficult examples

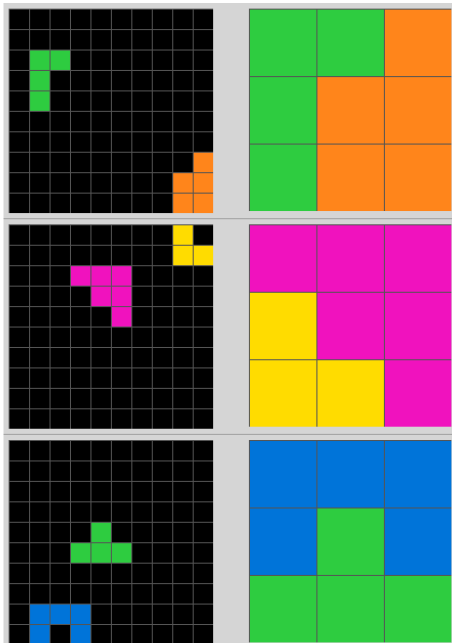


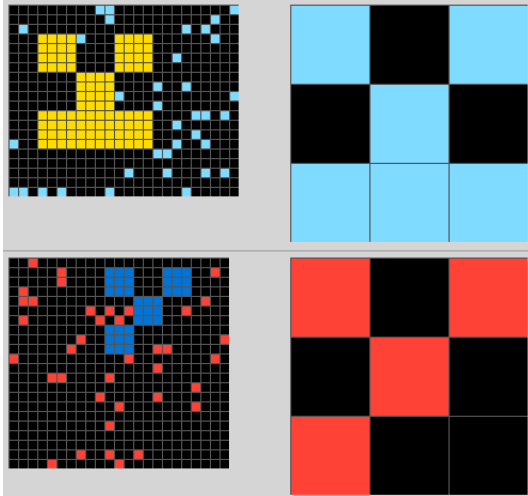
3.



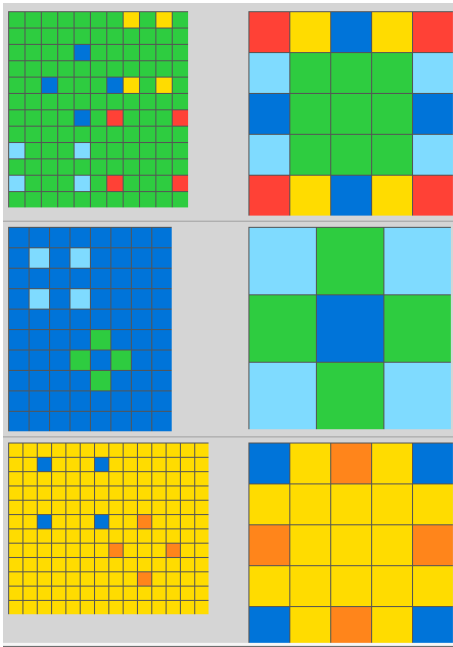
### Difficult examples

1.





2.



3.