

AI Programming

Week Eight

Errors & Debugging

Richard Price

rmp@cs.bham.ac.uk

www.cs.bham.ac.uk/~rmp/

www.cs.bham.ac.uk/internal/courses/ai-prog-a/



UNIVERSITY OF
BIRMINGHAM

Recap

- **Program design:**
 - Simple.
 - Plan.
 - Dynamic.
 - Flexible.
- **Algorithms.**

Types of Error

- **Three types of error:**
 - **Syntax errors.**
 - **Run-time errors.**
 - **Semantic errors.**

Syntax Errors

```
define printItemThree(list) -> output  
list(3) -> output;  
enddefine;  
printItemThree([1 2 3]) =>
```

```
;;; MISHAP ids: INCORRECT DEFINE SYNTAX (missing ; after header)  
;;; (At or before line 2 column 10)
```

- Type `help ids (Incorrect Define Syntax)` for further information.

Run Time Errors:

- The program compiles.
- But crashes when ran.

```
define printItemThree(list) -> output;  
list(3) -> output;  
enddefine;  
printItemThree([1 2]) =>
```

```
;;; MISHAP - BAD ARGUMENTS FOR INDEXED LIST ACCESS  
;;; INVOLVING: 3 [1 2]  
;;; FILE : /home/pg/mer/week8-lecture.p LINE NUMBER: 2  
;;; DOING : runproc
```

Semantic Errors

- Program compiles.
- Doesn't crash.
- Does not work as intended.
 - Incorrect output.
 - Infinite loops.
- Hardest problems to find and resolve.

Error Prevention

- **Make mistakes and learn from them.**
- **Follow your design.**
- **Modules**
 - **Break down code into procedures.**
 - **Ensure each procedure works.**
 - **Test cases.**
- **Layout of code.**
- **Test your code.**

Error detection

- Know your code.
- Print out your variables.
- Change one thing at a time.
- Use trace.

Trace

```
Define squarelt (input ) -> result.
```

```
    input *input -> result;
```

```
Enddefine;
```

```
Define squareList (inputList ) -> result;
```

```
Lvars element;
```

```
[%
```

```
    for element in inputList do
```

```
        squarelt (element );
```

```
    endfor
```

```
% ] -> result;
```

```
Enddefine;
```

```
Trace squarelt;
```

```
squareList ([4 6 2 8 99])=>
```

Trace

> squarelt 4
< squarelt 16
> squarelt 6
< squarelt 36
> squarelt 2
< squarelt 4
> squarelt 8
< squarelt 64
> squarelt 99
< squarelt 9801
**[16 36 4 64 9801]

Built in Help

- Three main levels of built-in help.
- **TEACH files**
 - Tutorial style information.
 - TEACH Teachfiles
- **Help files**
 - more concise information often with code examples.
 - HELP Helpfiles
- **REF files**
 - Contain reference only information.
 - REF Reffiles.

Mini Projects

- After this week you will be working on mini-projects.
- Worth 50% of your overall mark for AI programming A.
- We'll be using the lab sessions to discuss and program your projects.
- Plan Deadline: Monday 1st December 5pm.
- More details in the following lectures.
- Aim is to develop a large program in Pop-11.
- There is another mini-project next semester.
 - Focuses on applying Artificial Intelligence.

Mini Projects

- Encourage you to pick your own.
- Suggestions:
 - Intelligent System interface.
 - Expert system.
 - Medical diagnosis problem.
 - Fault diagnosis problem.
 - Football (or any sport) game predictor.
- Puzzle solvers
 - Sudoku
 - Crosswords.
- Board games
 - Connect 4.
 - Draughts
- ASCII games:
 - You wake up in a dark room, in front of you is a piece of cheese...