Ludic Computing Course CO345

Tutorial 1

Evolutionary Design

Simon Colton & Alison Pease
Computational Creativity Group
Department of Computing
Imperial College, London

ccg.doc.ic.ac.uk

Question 1

1. What tree and program fragment would you get if you if you randomly generated a tree in the following way:

a. start with an X

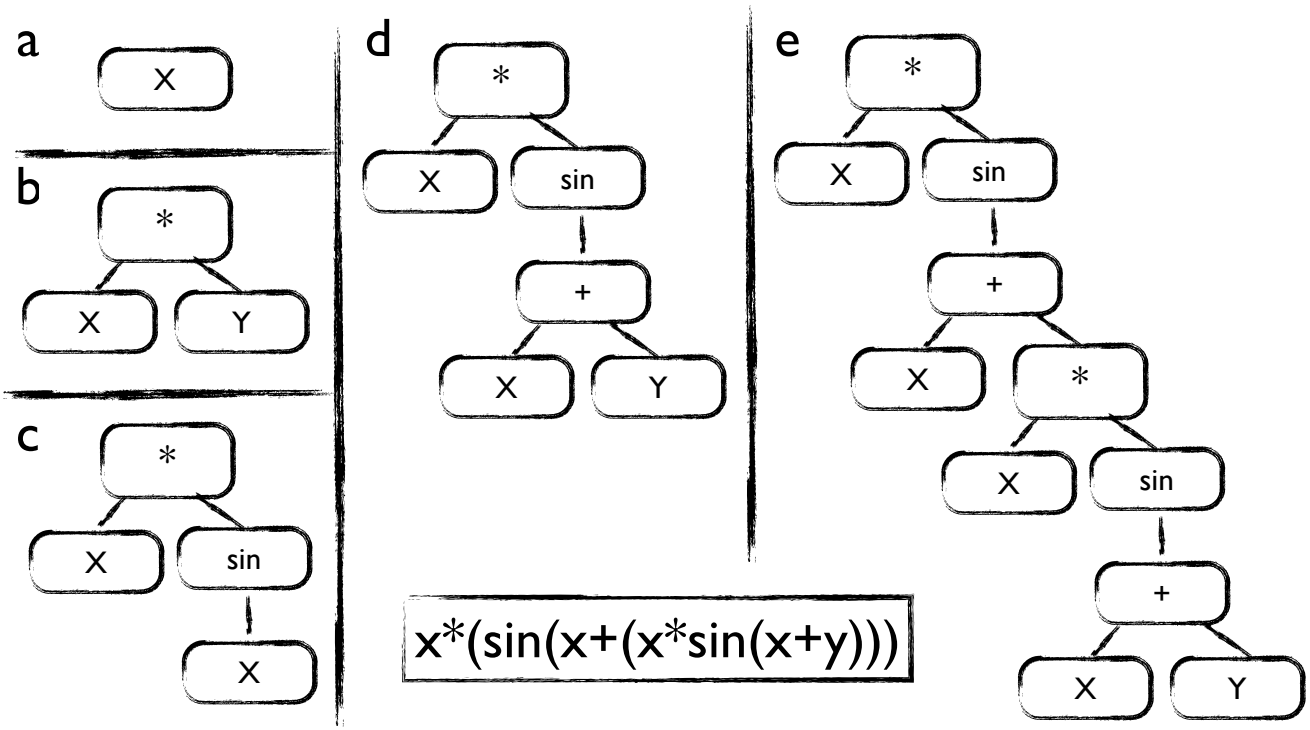
b. replace it with X*Y

c. replace the Y with sin(X)

d. replace the most recent X with X+Y

e. replace the Y with the entire tree so far

Answer 1



Questions 2 and 3

Suppose you have 7 individuals in a population, with the following fitness calculations:
$f(i_1)=20$, $f(i_2)=10$, $f(i_3)=5$, $f(i_4)=8$, $f(i_5)=1$, $f(i_6)=17$, $f(i_7)=3$

2. Write down a plausible intermediate population. Justify your answer

3. What would be the nature of the only tournament that individual i_7 would be selected from?

Answers 2 and 3

1. We first need to calculate the average fitness, which is: (20 + 10 + 5 + 8 + 1 + 17 + 3)/7 = 9.14

Then we have to work out $f(i_j)/9.14$ for each j and work out how many guaranteed IP entries each individual has, and what the probability P of gaining another one is.

$f(i_1)/9.14$ = 20/9.14 = 2.18 = 2 guaranteed, P = 0.18, 2 in IP is plausible

$f(i_2)/9.14$ = 10/9.14 = 1.09 = 1 guaranteed, P = 0.09, 1 in IP is plausible

$f(i_3)/9.14$ = 5/9.14 = 0.55 = 0 guaranteed, P = 0.55, 1 in IP is plausible

$f(i_4)/9.14$ = 8/9.14 = 0.86 = 0 guaranteed, P = 0.86, 1 in IP is plausible

$f(i_5)/9.14$ = 1/9.14 = 0.11 = 0 guaranteed, P = 0.11, 0 in IP is plausible

$f(i_6)/9.14$ = 17/9.14 = 1.86 = 1 guaranteed, P = 0.86, 2 in IP is plausible

$f(i_7)/9.14$ = 3/9.14 = 0.33 = 0 guaranteed, P = 0.33, 0 in IP is plausible

Hence, a plausible IP is: { i_1, i_1, i_2, i_3, i_4, i_6, i_6 }

2. i_7 is only fitter than i_5, so the tournament would have to be for two individuals, with i_7 competing against i_5

Questions 4, 5 and 6

4. Which of the individuals, U,V or W achieves the correct input/output pairings for the given program? Justify your answer.

Which parameter could be mutated without any change to the program I/O?

5. Perform 1-point crossover on individuals V and W to generate a program which outputs a different value for input (17,17) than both its parents

6. Draw the body of the program with parameters U in a suitable tree representation

```
public float tutorial(int x, int y){
  if (x < A * y) return x + y;
  else if (y < 10 * Math.sqrt(x)) {
    return y + Math.abs(x - B);
  }
  else return C * x;
}
```

```
U: A=0, B=3,  C=5
V: A=1, B=2,  C=3
W: A=2, B=10, C=2
```

```
In=(9,8)   Out=17
In=(1,2)   Out=3
In=(16,2)  Out=8
```

Answer 4

```
public float tutorial(int x, int y){
   if (x < 0 * y) return x + y;
   else if (y < 10 * Math.sqrt(x)) {
     return y + Math.abs(x - 3);
   }
   else return 5 * x;
}
```

```
tutorial(9,8)=14
tutorial(1,2)=4
tutorial(16,2)=15
```

U

```
public float tutorial(int x, int y){
   if (x < 1 * y) return x + y;
   else if (y < 10 * Math.sqrt(x)) {
     return y + Math.abs(x - 2);
   }
   else return 3 * x;
}
```

```
tutorial(9,8)=15
tutorial(1,2)=3
tutorial(16,2)=16
```

V

```
public float tutorial(int x, int y){
   if (x < 2 * y) return x + y;
   else if (y < 10 * Math.sqrt(x)) {
     return y + Math.abs(x - 10);
   }
   else return 2 * x;
}
```

```
tutorial(9,8)=17
tutorial(1,2)=3
tutorial(16,2)=8
```

W ✔

Final clause of the conditional is never used
Hence parameter C could be mutated without any effect on the output for these inputs

Answer 5

```
public float tutorial(int x, int y){
   if (x < 1 * y) return x + y;
   else if (y < 10 * Math.sqrt(x)) {
     return y + Math.abs(x - 2);
   }
   else return 3 * x;
}
```

```
tutorial(17,17) = 32
```

V

```
public float tutorial(int x, int y){
   if (x < 2 * y) return x + y;
   else if (y < 10 * Math.sqrt(x)) {
     return y + Math.abs(x - 10);
   }
   else return 2 * x;
}
```

```
tutorial(17,17) = 34
```

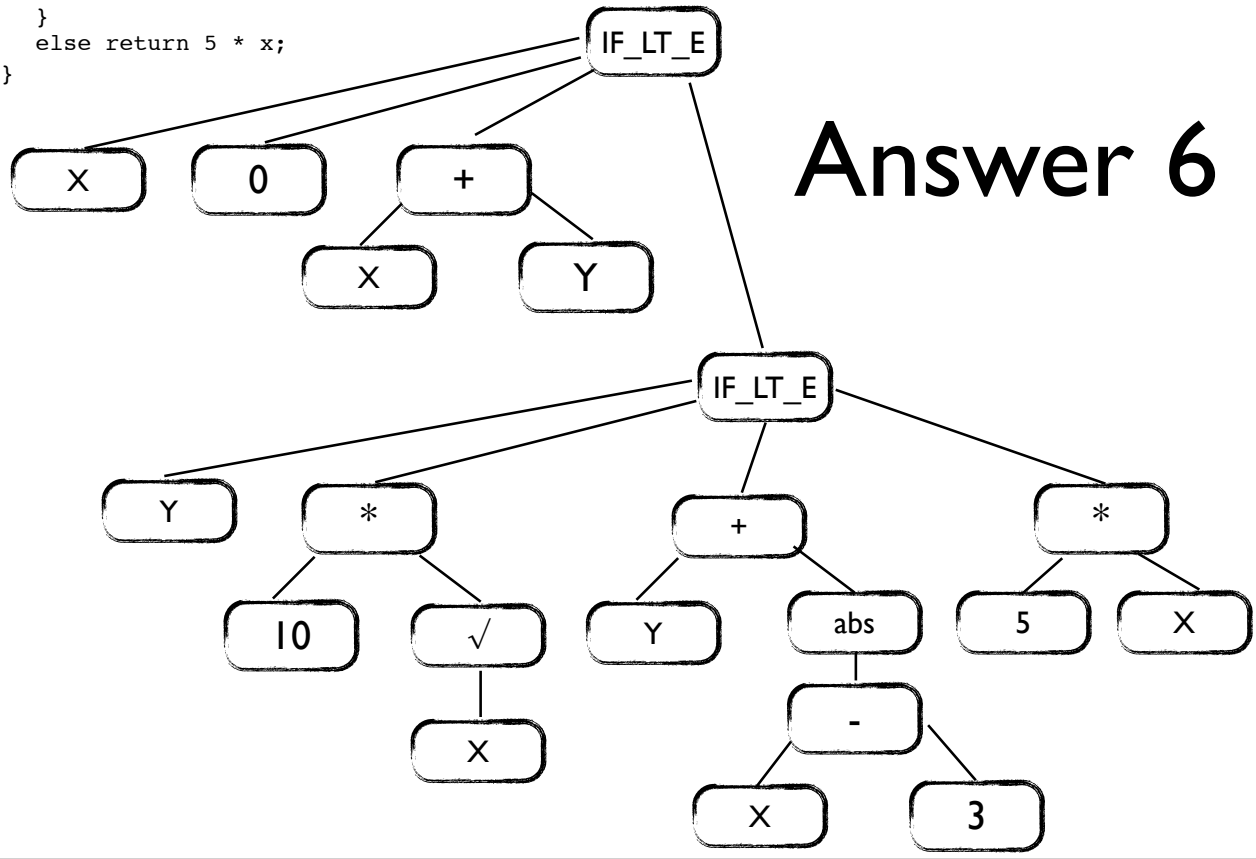
W

Crossover [1,2,3] and [2,10,2] to give [1,10,2]

```
public float tutorial(int x, int y){
   if (x < 1 * y) return x + y;
   else if (y < 10 * Math.sqrt(x)) {
     return y + Math.abs(x - 10);
   }
   else return 2 * x;
}
```

```
tutorial(17,17) = 24
```

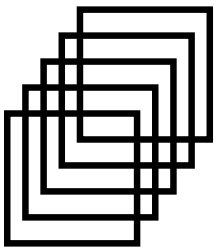
```
public float tutorial(int x, int y){
   if (x < 0 * y) return x + y;
   else if (y < 10 * Math.sqrt(x)) {
      return y + Math.abs(x - 3);
   }
   else return 5 * x;
}
```

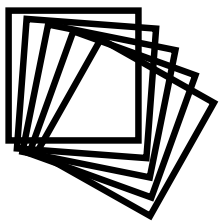


Answer 6

Questions 7 and 8

7. Suppose you wanted to evolve programs which can produce spirographs by changing a unit square's position and rotation over a hundred time steps, t, and rendering each square. Write down a suitable wrapper program.





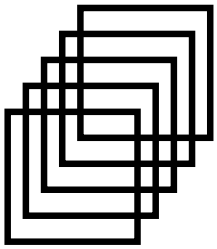
8. Using the wrapper, fill in a program which might approximately produce each of the spirographs above

Answer 7

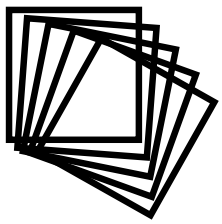
The wrapper program has to take in the time step, t, then set a rotation angle and a co-ordinate for the square and finally draw the square. The following would be suitable:

```
int r = 0;
int x = 0;
int y = 0;

get_square(int t){
  // update r based on t
  // update x and y based on t
  drawSquare(x,y,r)
}
```

Answer 8



```
int r = 0;
int x = 0;
int y = 0;

get_square(int t){
  x = 3 * t;
  y = 0 - (3 * t);
  drawSquare(x,y,r)
}
```



```
int r = 0;
int x = 0;
int y = 0;

get_square(int t){
  r = 10*t;
  x = 2*t;
  y = t;
  drawSquare(x,y,r)
}
```