**Presentatie**

Inleiding

- Voorstellen  
*Deel 1: case uitleggen*  
- Wij hebben de fruitvlieg case  
- De genen van de Drosophila Melanogaster en de Drosophila Miranda zijn identiek, maar staan in een andere volgorde

- Het is mogelijk dat het ene genoom in het andere verandert d.m.v. mutaties  
- Aan ons de taak om uit te zoeken hoe je van het ene genoom naar het andere genoom komt  
- Kan meerdere manieren, maar het kortste pad / het kleinste pad is het waarschijnlijkst  
- Daarom: was onze opdracht 🡪 vind het korste / kleinste pad!

- de rest van de opdrachten  
  
*Deel 2: toestandsruimte/zoekruimte uitleggen*

Het genoom van fruitvlieg bestaat uit 25 genen.

n = lengte van de lijst  
Zoekruimte = (Σ n – 1 ) diepte

Methode  
- Van genen overgestapt naar getallen (handiger om mee te werken)

- Uitleggen wat niet werkt en waarom het niet werkt  
 \* keuze tussen sorteer- of zoekalgoritme   
 \* depth first, breadth first 🡪 toestandsruimte te groot  
 \* iterative deepening 🡪 beperken, maar bleek alsnog te groot te zijn

- Wat hebben we dan wel gebruikt?

\* We wilden “A-star” 🡪 net iets anders geworden, niet onderschatten

- Breakpoints uitleggen  
 \* waarom werkt dit wel?  
- Uitleggen wat er gebeurt bij verschillende cases  
 \* Stoppen na 5 minuten als het niet werkt, dan overstappen naar andere mogelijkheid

Resultaten

Case a: kortste pad van gegeven genoom 🡪 13 stappen + tijd +hoeveel dingen checkt ie

Case b: min, max, gemiddelde, SD (van stappen en tijd + hoeveel dingen checkt ie)

Case c: kleinste pad 🡪 88 (?) + tijd + hoeveel dingen checkt ie

Case d: min, max, gemiddelde, SD (van stappen en tijd + hoeveel dingen checkt ie)

Conclusie en Discussie  
  
Case e: genoom met veel breakpoints veel mutatiestappen nodig om gesorteerd te worden. Het maakt uit of veel delen al naast elkaar staan en of buren al beetje goed staan.  
  
Case f:  
  
Case g: