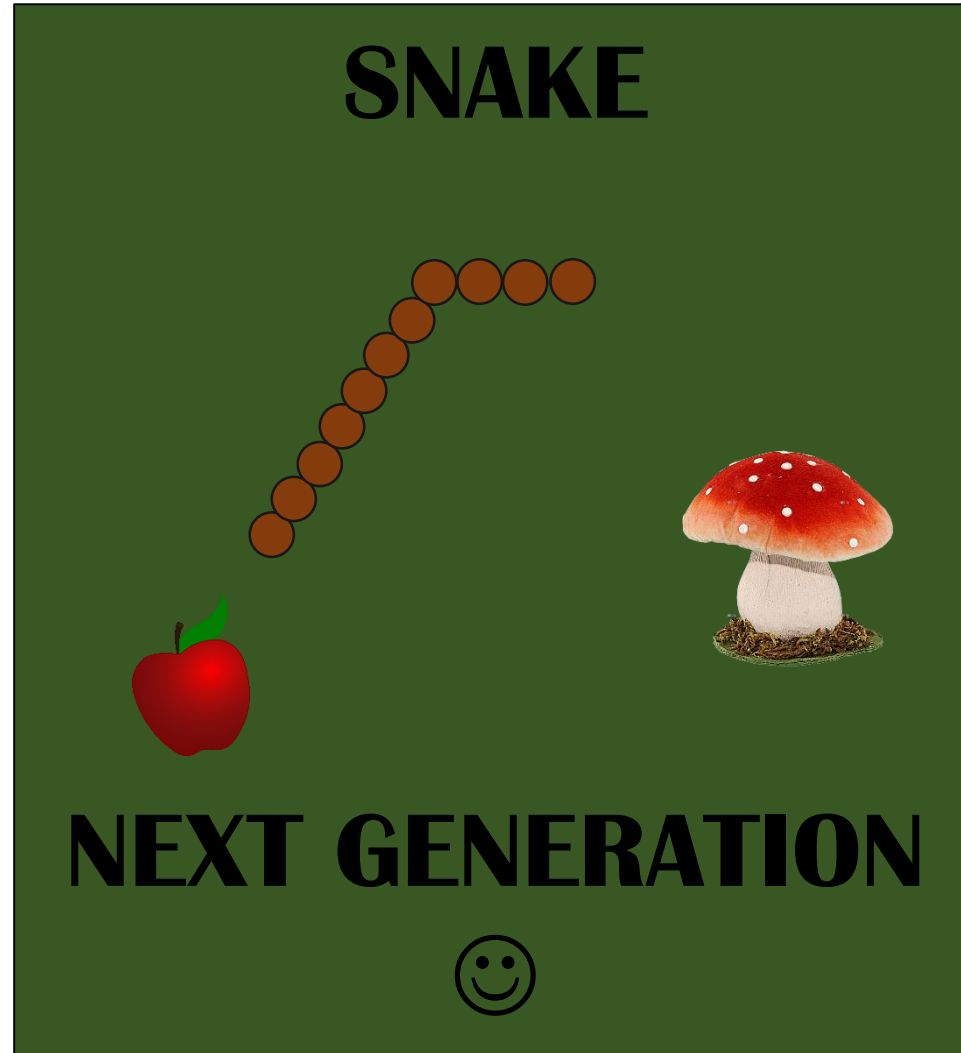


DATASTRUKTURER OCH ALGORITMER:

Projekt 1



PROJEKT BESKRIVNING


- Gör ett spel för Android som fungerar enligt det klassiska spelet "Snake"
- I denna variant av spelet ska man dock kunna styra i vilken riktning som helst (inte enbart horisontalt eller vertikalt)
- Spelet går ut på att masken ska äta så många äppel som möjligt utan att krocka med sej själv, någon av kanterna eller med en flugsvamp (då tar spelt genast slut).
- För varje gång man äter ett äppel, växer masken.
- Enbart ett äppel åt gången syns i spelrutan och när det blir uppätet kommer ett nytt äppel in i spelrutan på en slumpmässig plats
- Antalet flugsvampar i spelrutan ökar under spelets gång (t.ex. endast en från början)
- Före man börjar spela ska man kunna skriva sitt namn och spelet upprätthåller en lokal "Highscore"-lista på maximala antalet äppel varje spelare har lyckats samla.

SPELETS VYER


Ditt namn:

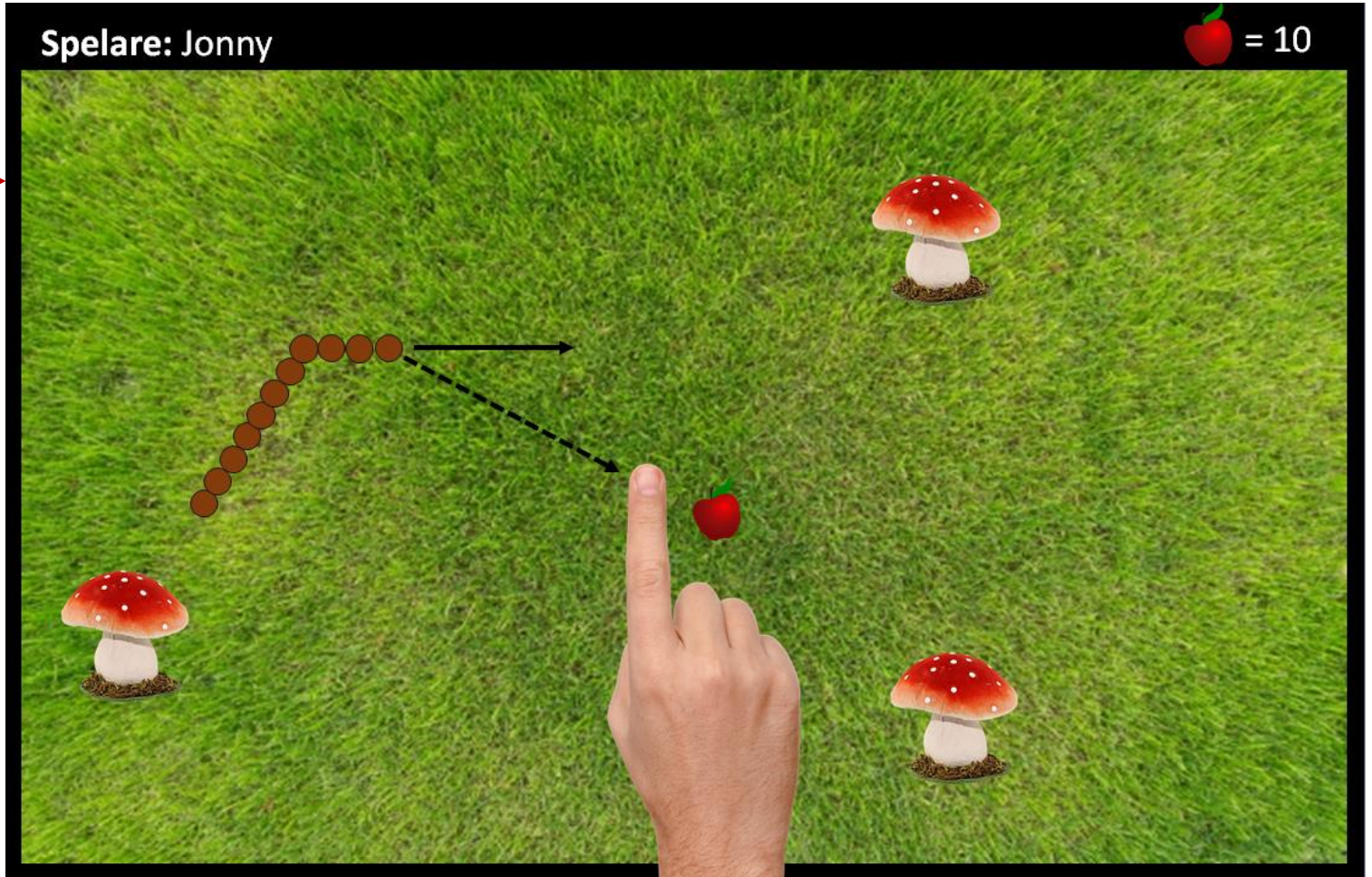
Jonny

Starta spelet

Spelare	
Jonny	999
Gunilla	80
Nisse	72
Lisa	50
Anna	44
Magnus	33

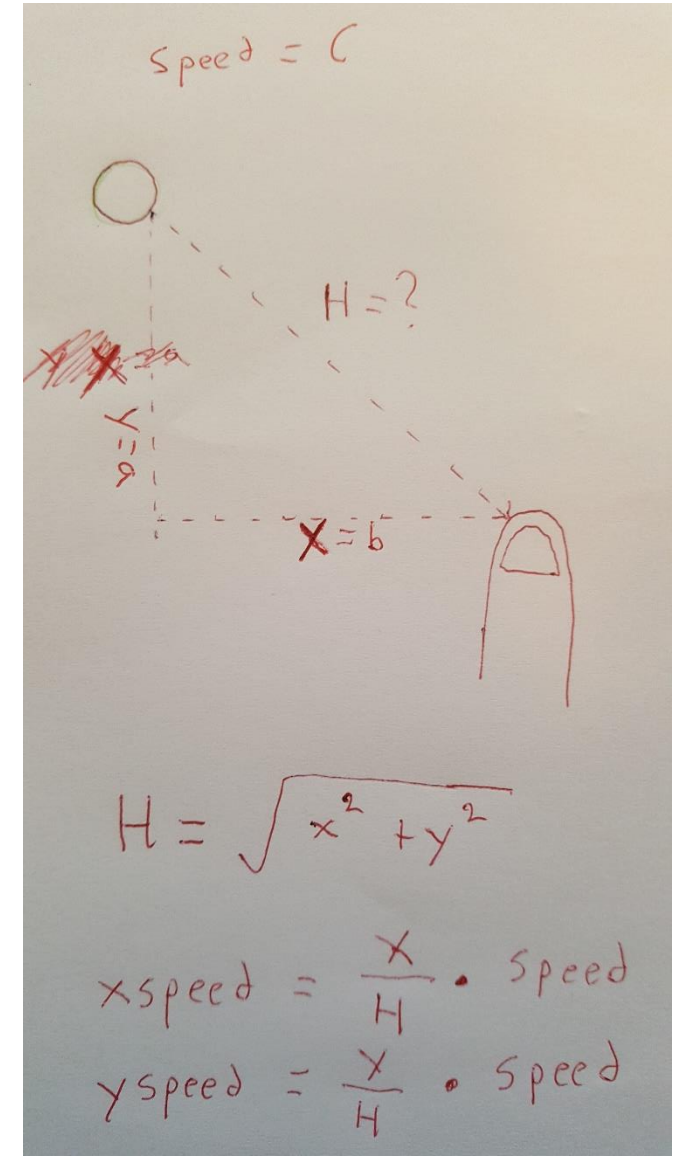
Spelare: Jonny

 = 10



DELMOMENT 1: Maskens styrning (11 p)

- Animera maskens "huvud" (kan vara en boll)
- När spelet startas rör sej masken i en viss fart ($Speed = C$) och i en viss riktning, t.ex $xspeed = Speed$, $yspeed = 0$ (dvs. rör sej rakt till höger)
- När man trycker med fingret någonstans på spelrutan ändrar maskens huvud riktning och börjar röra sej mot det ställe man tryckte ner fingret
- Nya värden för $xspeed$ och $yspeed$ kan då beräknas enligt bilden till höger

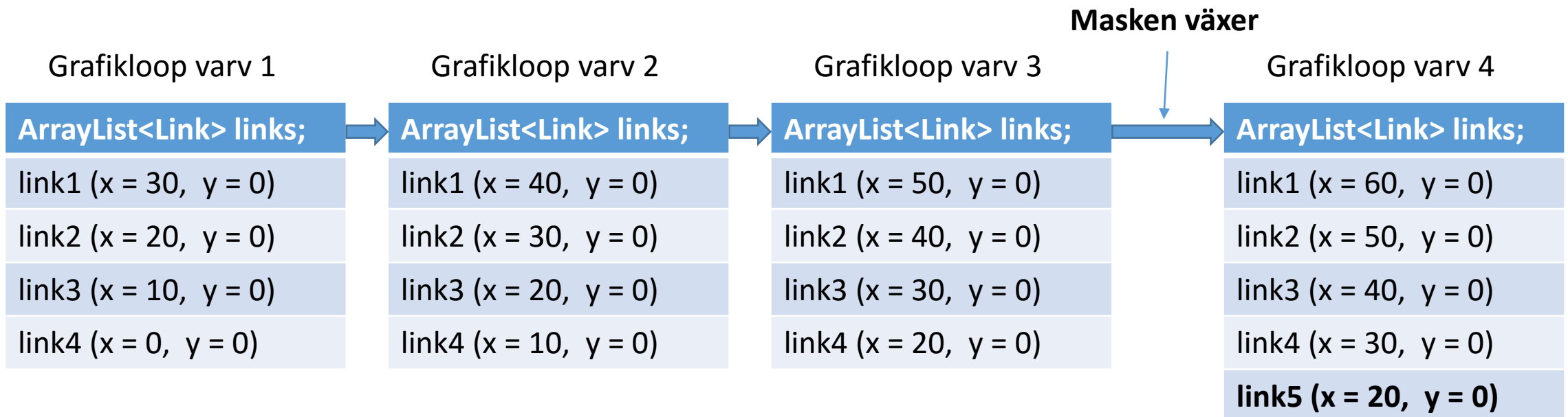


DELMOMENT 2: Maskens "svans" (13 p)

- Skapa en klass *Link* som definierar bl.a. bredd, höjd, x-position, y-position, *Speed*, *xspeed* och *yspeed* för en länk (boll) av masken.
- Skapa t.ex. en *ArrayList* som du kallar för *links* eller *snake* dit du sparar alla maskens länkar, t.ex. så att huvud är först i listan och ändan på svansen sist i listan.
- När masken växer, sätt in en ny länk i slutet av listan vars position blir samma som föregående länks position efter att alla övriga länkar har flyttat ett steg framåt

DELMOMENT 2: Maskens "svans" (13 p)

- Länk nr n får alltså samma position som länk nr $n-1$ hade på föregående loop-varv (i grafik-loopen).
- I exemplet nedan rör sej masken rakt till höger ($xspeed = 10, yspeed = 0$)



DELMOMENT 3: Krockar (6 p)

- Sätt in "krockcheckar" som gör att själva spelet avslutas och användaren styrs tillbaka till startvyn om maskens huvud rör antingen någon kant eller en flugsvamp
- Flugsvamparna placeras ut på slumpmässiga positioner i spelområdet på så sätt att när spelet börjar finns det bara en flugsvamp, men vartefter spelet pågår slumpas det in flere flugsvampar för att göra spelet svårare.

DELMOMENT 4: Ätfunktion (10 p)

- När masken rör vid ett äppel skall
 - masken växa, t.ex. så att det kommer en länk till på slutet
 - spelaren få +1 poäng. Antalet poäng som spelaren samlat, dvs. antalet äppel som masken ätit skall synas någonstans på spelytan
 - äppet få en ny slumpmässig position på spelytan

DELMOMENT 5: Grafik (8 p)

- Sätt in grafik (rita ut Bitmaps) för:
 - Bakgrunden
 - Äpplet
 - Flugsvamparna
- Använd t.ex. denna video som hjälp:
https://www.youtube.com/watch?v=slgyU7Mlt_0

DELMOMENT 6: Highscore (12 p)

- Spelet skall hålla reda på alla namn som registreras när man börjar spelet
- När spelet är slut skall appen kontrollera om spelaren har spelat tidigare och i så fall uppdatera poängrekordet för den spelare om de nya poängen är högre än de gamla. Om spelaren inte finns från tidigare i listan, tillsätts spelaren samt poängen till listan.
- Highscore-listan skall alltid synas i starvtyn och den ska vara sorterad enligt antal poäng på så vis att den spelare som fått mest poäng visas först i listan.

SAMMANDRAG ÖVER BEDÖMNINGEN

Delmoment	Poäng
Maskens styrning (enbart huvudet)	11
Maskens svans (resten av "kroppen" följer huvudet)	13
Krockar (mot väggar och flugsvampar)	6
Ätfunktion	10
Grafik (bitmaps)	8
Highscore	12
TOTALT	60