REPORT TO BOSS

(Xinqian Wang s4565489/Ziyi Liu 45655335)

Formulation

Communication10

Data: *SpeciesMap* — Dictionary contains label site and its value for which species live in the site.

Stages: The stages are in the state.

State: *SavedSpecies* — List contains the species are saved.

Actions: Restoring sites to save species.

Let $f(SavedSpecies, k)$ be equal to SavedSpecies with list k contains sites being restored.

Value Function: Minimise the length of k sites until we get $SavedSpecies$ with the length of 20.

Contribution Function: Calculate the number of species are saved.

We want $f(SavedSpecies = [\,], k = [\,])$.

$$\begin{array}{c} f(SavedSpecies, k) \\ i \in SpeciesMap \\ if\ i\ not\ in\ k\ and\ len(k) < 20 \end{array} = \min\{\, f(SavedSpecies + SpeciesMap[i], k + [i])\, \}$$

$$\begin{array}{c} f(SavedSpecies, k) \\ if\ len(k) = 20 \end{array} = (len(k), SavedSpecies, k)$$

Communication11

Data: $\underline{Size_i}$ — The size of species $i \in [0,6,7,9,16]$.

$\underline{Value_i}$ — The value of species $i \in [0,6,7,9,16]$.

Stages: The stages are in the state.

State: Capacity — The capacity of the container left.

Actions: Packing species i into the container if the Capacity has space.

Value Function: Maxmise the total value of container until it has been full.

We want $f(\text{Capacity} = 0)$.

$$\begin{array}{c} f(\text{Capacity}) \\ i \in [0,6,7,9,16] \\ if\ \text{Capacity} \geq Size_i \end{array} = \max\{\, Value_i + f(\text{Capacity} - Size_i)\, \}$$

$$\begin{array}{c} f(\text{Capacity}) \\ i \in [0,6,7,9,16] \\ if\ \text{Capacity} < \min\{\, Size_i\, \} \end{array} = 0 + f(\text{Capacity})$$

Communication12

Data: $\underline{AvailableResort}$ List of saving species in sequence.

$\underline{LostProb}$ List contains the lost probability for each site.

Stages: The stages are in the state.

State: $\underline{SD}$ List contains the current statue which represents site in

$$AvailableResort. \underset{i \in SD}{i} = \begin{cases} -1\, if\ \ site\ \ i\, is\, lost \\ 0\, if\ \ site\ \ i\, is\, fragile \\ 1\, if\ \ site\ \ i\, is\, restored \end{cases}$$

Actions: Restoring the fragile sites from the *SD* sequentially.

Contribution Function: $\underline{C(SD, LostProb)}$

Calculate and return a dictionary contains label of the probabilitie and its state.

$\underline{VN(SD)}$

Calculate the number of species have been saved, according to $AvailableResort.$

Value Function: Calculate the expected number of species been saved.

We want $f(SD = [\ 0,0,0,0,0,0,0,0,0\ ])$.

$$\underset{if\ 0\ in\ SD}{f(SD)} = \underset{for\ j\ in\ C(SD, LostProb)}{\text{sum}} \{\, j[0] * f(SD = j[1])\, \}$$

$$\underset{if\ 0\ not\ in\ SD}{f(SD)} = VN(SD)$$

Communication13

Data: <u>SpeciesMap</u> Dictionary contains label site and its value for which species live in the site.

<u>LostProb</u> List contains the lost probability for each site.

Stages: The stages are in the state.

State: <u>SD</u> List contains the current statue which represents site in SpeciesMap. $\underset{i\ \in\ SD}{i} = \begin{cases} -1\, if\ \ site\ \ i\, is\, lost \\ 0\, if\ \ site\ \ i\, is\, fragile \\ 1\, if\ \ site\ \ i\, is\, restored \end{cases}.$

Example: $SD[0]$ represents the first site which is *SpeciesMap*[0].

Actions: Restoring the fragile sites from the *SD*.

Contribution Function: $\underline{C(SD, LostProb)}$

Calculate and return a dictionary contains label of the probabilitie and its state.

$\underline{UPDATE\ SDL(SD)}$

Return the tuple which contains three lists, Saved, Dead and Left.

$$\underset{for\ i\ in\ SD\ if\ i\ ==\ 1}{Saved} = [SpeciesMap[i]\ ]$$

$$\underset{for\ i\ in\ SD\ if\ i\ ==\ -1}{Dead} = [SpeciesMap[i]\ ]$$

$$\underset{for\ i\ in\ SD\ if\ i\ ==\ 0}{Left} = [SpeciesMap[i]\ ]$$

$\underline{VN\ 13(Saved)}$

Calculate the number of species have been saved, according to $SpeciesMap$.

$\underline{SavedWho(Saved, Left)}$

Return a list of SD which represents the possible outcomes of restoring.

Value Function: Maximise the expected number of species been saved.

We want $f(SD = [\ 0,0,0,0,0,0,0,0,0\ ])$.

$$\underset{if\ 0\ in\ SD}{f(SD)} = \underset{\substack{for\ SD^*\ in\ SavedWho(Saved, Left) \\ \text{Saved}=\ Update\_SDL(SD)[0] \\ \text{Left}=\ Update\_SDL(SD)[2]}}{\max} \left\{ \underset{for\ j\ in\ C(SD^*, LostProb)}{\text{sum}} \{\, j[0] * f(SD = j[1])\ \} \right\}$$

$$\underset{if\ 0\ not\ in\ SD}{f(SD)} = \underset{\substack{\text{Saved}=\ Update_{SDL(SD)[0]} \\ \text{Left}=\ Update_{SDL(SD)[2]}}}{\text{VN\_13}} (Saved)$$

Communication14

Data: SpeciesMap — Dictionary contains label site and its value for which species live in the site.

LocationMap — Dictionary contains label site and its adjacent sites.

LostProb — List contains the lost probability for each site.

Stages: The stages are in the state.

State: SD — List contains the current statue which represents site in SpeciesMap. $\underset{i \in SD}{i} = \begin{cases} -1\ if\ \ site\ \ i\ is\ lost \\ 0\ if\ \ site\ \ i\ is\ fragile\ . \\ 1\ if\ \ site\ \ i\ is\ restored \end{cases}$

Example: $SD[0]$ represents the first site which is *SpeciesMap*[0].

Actions: Restoring the fragile sites from the *SD*.

Contribution Function: $\underline{C(SD, LostProb)}$

Calculate and return a dictionary contains label of the probabilitie and its state.

$\underline{LostRate(SD)}$

Calculate the lost probability of each site according to the sites which have been lost and their location.

$\underline{UPDATE\ SDL(SD)}$

Return the tuple which contains three lists, Saved, Dead and Left.

$$\underset{for\ i\ in\ SD\ if\ i == 1}{Saved} = [SpeciesMap[i]\ ]$$

$$\underset{for\ i\ in\ SD\ if\ i == -1}{Dead} = [SpeciesMap[i]\ ]$$

$$\underset{for\ i\ in\ SD\ if\ i == 0}{Left} = [SpeciesMap[i]\ ]$$

$\underline{VN\ 13(Saved)}$

Calculate the number of species have been saved, according to $SpeciesMap$.

$\underline{SavedWho(Saved, Left)}$

Return a list of SD which represents the possible outcomes of restoring.

Value Function: Maximise the expected number of species been saved.

We want $f(SD = [\ 0,0,0,0,0,0,0,0,0\ ])$.

$$\underset{if\ 0\ in\ SD}{f(SD)} = \underset{\substack{for\ SD^*\ in\ SavedWho(Saved, Left) \\ \text{Saved} = Update\_SDL(SD)[0] \\ \text{Left} = Update_{SDL(SD)[2]} \\ LostProb = LostRate(SD)}}{\max} \left\{ \underset{for\ j\ in\ C(SD^*, LostProb)}{\text{sum}} \{\ j[0] * f(SD = j[1])\ \} \right\}$$

$$\underset{if\ 0\ not\ in\ SD}{f(SD)} = \underset{\substack{\text{Saved} = Update_{SDL(SD)[0]} \\ \text{Left} = Update_{SDL(SD)[2]}}}{\text{VN\_13}} (Saved)$$

Communication15

Data: SpeciesMap — Dictionary contains label site and its value for which species live in the site.

LocationMap — Dictionary contains label site and its adjacent sites.

$LostProb$ — The list contains the lost probability for each site.

TouristList — The list contains the species that the tourists really come to see.

Stages: The stages are in the state.

State: SD — List contains the current statue which represents site in

$$\text{SpeciesMap.} \underset{i \in SD}{i} = \begin{cases} -1\, if\ \ site\ \ i\, is\, lost \\ 0\, if\ \ site\ \ i\, is\, fragile \\ 1\, if\ \ site\ \ i\, is\, restored \end{cases}.$$

Example: $SD[0]$ represents the first site which is *SpeciesMap*[0].

Actions: Restoring the fragile sites from the *SD*.

Contribution Function: $\underline{C(SD, LostProb)}$

Calculate and return a dictionary contains label of the probabilitie and its state.

$\underline{LostRate(SD)}$

Calculate the lost probability of each site according to the sites which have been lost and their location.

$\underline{UPDATE\ SDL(SD)}$

Return the tuple which contains three lists, Saved, Dead and Left.

$$\underset{for\ i\ in\ SD\ if\ i == 1}{Saved} = [SpeciesMap[i]\ ]$$

$$\underset{for\ i\ in\ SD\ if\ i == -1}{Dead} = [SpeciesMap[i]\ ]$$

$$\underset{for\ i\ in\ SD\ if\ i == 0}{Left} = [SpeciesMap[i]\ ]$$

$\underline{VN\_15(Saved)}$

Return the list which contains species have been saved, according to $SpeciesMap$.

$$\underset{\forall j\ \in\ TouristList}{\text{VN\_15(Saved)}} = \begin{cases} 1, & if\ j\ in\ Saved \\ 0, & if\ j\ not\ in\ Saved \end{cases}$$

$\underline{SavedWho(Saved, Left)}$

Return a list of SD which represents the possible outcomes of restoring.

Value Function: Maximise the expected number of species been saved.

We want $f(SD = [\ 0,0,0,0,0,0,0,0,0\ ])$.

$$\underset{if\ 0\ in\ SD}{f(SD)} = \underset{\substack{for\ SD^*\ in\ SavedWho(Saved, Left) \\ \text{Saved} = Update\_SDL(SD)[0] \\ \text{Left} = Update_{SDL(SD)[2]} \\ LostProb = LostRate(SD)}}{\max} \left\{ \underset{for\ j\ in\ C(SD^*, LostProb)}{\text{sum}} \{\, j[0] * f(SD = j[1])\ \} \right\}$$

$$\underset{if\ 0\ not\ in\ SD}{f(SD)} = \underset{\substack{\text{Saved}= Update_{SDL(SD)[0]} \\ \text{Left}= Update_{SDL(SD)[2]}}}{\text{VN\_15}} (Saved)$$