Complete

ToC

DBMS

Revision

AI LAST MINUTE NOTES

CombineCS
The Extra Step

Public Notice
Dated: 03.09.2021

Rescheduling of the Examination dates for UGC-NET December 2020 and June 2021 cycles

The examination for the **University Grants Commission (UGC)-NET December 2020 and June 2021 cycles** is scheduled from **06 October to 11 October 2021**.

NTA has come to know from the student community that the **October 10 examination date is clashing** with some major Examinations that also have been scheduled for **that day.** With a view to ensure larger participation of candidates and to remove the hardship caused to them, it has been decided to **reschedule some of the dates of the UGC-NET December 2020 and June 2021 cycles.**

The revised schedule is as follows:

6,7,8 ,17,18,19 Exam date

Old days and dates	Revised days and dates
Wednesday 06 October to Monday 11 October 2021	Wednesday 06 to Friday 08 October 2021 and Sunday 17 to Tuesday 19 October 2021

The detailed date sheet will be given subsequently.

The candidates are advised to **keep visiting the official website of NTA** (ugcnet.nta.nic.in) and (www.nta.ac.in) for the latest updates.

https://t.me/RashmiCCS

My class @ 3:30pm daily on YouTube

→ Subscribe

→ enroll now for 1-to-1 doubt solution

SUBSCRIBE

One Liner Notes

Uninformed Search

alternative names

1. **Breadth-first Search.**
2. **Depth-first Search.**
3. Depth-limited **Search.**
4. Iterative deepening **depth-first search.**
5. Uniform cost **search.**
6. Bidirectional **Search**

minimum cost

Informed Search

heuristic

guess

1. **Best First Search Algorithm(Greedy/Recursive search)**
2. **A* Search Algorithm** – 2020
3. **Recursive best-first search**
4. **AO* Search Algorithm**
5. Hill Climbing → 2020
6. Genetic Algorithm
7. SMA* → 2020

Uninformed Properties	DFS	BFS	Bidirectional	Uniform Cost	Depth Limit	Iterative Deepening
Complete	NO	YES	YES	YES	NO	YES
Optimal	NO	YES	YES	YES	NO	YES
Time Complexity	$O(b^m)$ linear	$O(b^d)$ Exponential	$O(b^{d/2})$	$O(b^d)$	$O(b^l)$	$O(b^d)$
Space Complexity = max depth	$O(bm)$	$O(b^d)$	$O(b^{d/2})$	$O(b^d)$ BFS = Uniform Cost	$O(bl)$ l = limit	$O(bd)$
Implement	STACK	QUEUE				

Informed Search Properties	Greedy Best Search	A* 2020	Recursive Best Search	SMA* Simplified Memory Bounded A*
Objective Function	f(n) = h(n)	f(n) = g(n) + h(n)	Same as A*	Same as A*
Complete	No (by DFS, can stuck in loops)	YES	YES	YES
Optimal	No (by BFS, all nodes traverse)	YES (Admissible & Consistent)	YES	YES
Time Complexity	O (b^d)	O (b^d) Exponential		
Space Complexity	O (b^d)	O (b^d) Keeps all nodes in memory	O (bd) linear	expected 2021
Implement	Priority Queue			

UGCNET DEC 2014

Q) An A* algorithm is a heuristic search technique which…?

1. **is like a depth-first search where most promising child is selected for expansion.**
2. **generates all successor nodes and computes an estimate of distance (cost) from start node to a goal node through each of the successors. It then chooses the successor with shortest cost.**
3. **saves all path lengths (costs) from start node to all generated nodes and chooses shortest path for further expansion. Branch and Bound**
4. **None of the above**

UGCNET DEC 2014

Q) An A* algorithm is a heuristic search technique which…?

1. **is like a depth-first search where most promising child is selected for expansion. Hill Climbing**
2. **generates all successor nodes and computes an estimate of distance (cost) from start node to a goal node through each of the successors. It then chooses the successor with shortest cost.**
3. **saves all path lengths (costs) from start node to all generated nodes and chooses shortest path for further expansion. Branch and Bound**
4. **none of the above**

BFS – Queue
DFS – STACK
Best First / Branch & Bound – Priority Queue

LMN

Hill Climbing – Problems & Solutions
a) Local Maxima - Backtracking
b) Plateau – Random / Big Jump
c) Ridge – Bi-directional

Branch & Bound (Tree/Graph) –
- ✓ keeps track of all partial path which can be candidate for further exploration.
- ✓ Optimization Problem
- ✓ Used when Greedy & Dynamic Programming Fails
- ✓ Slow method
- ✓ Exponential Complexity

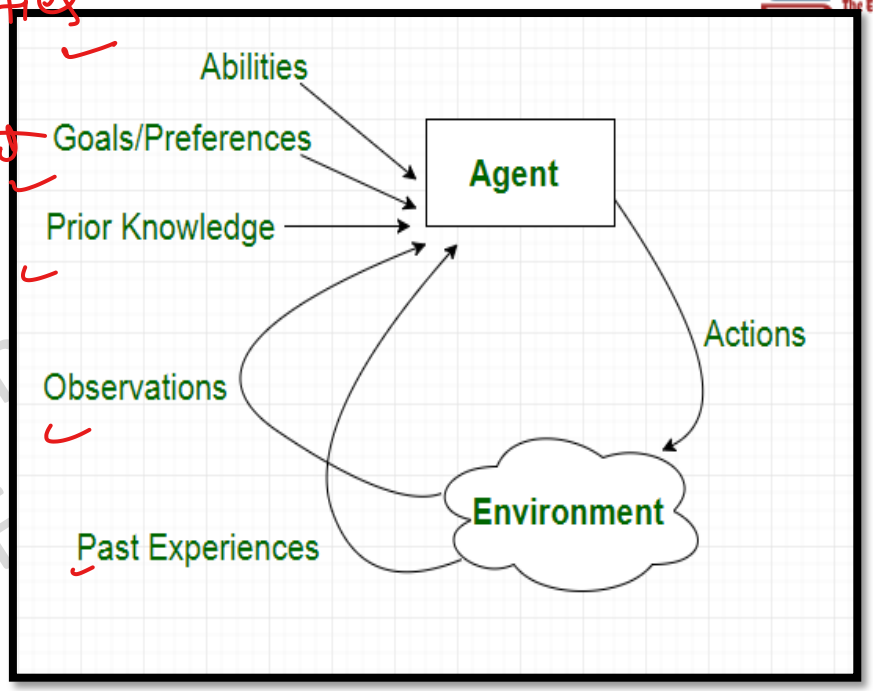
Hill Climbing – Heuristic Search Algorithm
- ✓ Solve Optimization Problem
- ✓ Heuristic – may not give optimal solution, but Good solution can be achieved in reasonable time.
- ✓ Heuristic – ALL Path trace
- ✓ Based on Greedy Approach
- ✓ 3 types

a) Simple - First
b) Steepest - ALL
c) Stochastic - Random

Acting Rationally:
The rational Agent approach

Agent is something that acts. Computer agent is expected to have following attributes:

a) Autonomous control
b) Perceiving their environment
c) Persisting over a prolonged period of time
d) Adapting to change
e) And capable of taking on another's goal



Task environments

2020 (19)

properties based Ques.

1. **Fully observable vs. partially observable**
2. **Deterministic vs. stochastic**
3. **Episodic vs. sequential**
4. **Static vs. dynamic**
5. **Discrete vs. continuous**
6. **Single agent vs. multi agent**

Types of Agent

Problems with Simple reflex agents are :

a) **Very limited intelligence.**

b) **No knowledge of non-perceptual parts of state.**

c) **Usually too big to generate and store.**

d) **If there occurs any change in the environment, then the collection of rules need to be updated.**

e) **Lacking history, easily get stuck in infinite loops.**

One solution is to randomize actions

Model-based reflex – (Knowledge base)

partially observable environments

keep track of internal state (Past history +present)

Goal-based –

Expansion of Model based Reflex Agents

focus on Goal state (Supervised learning)

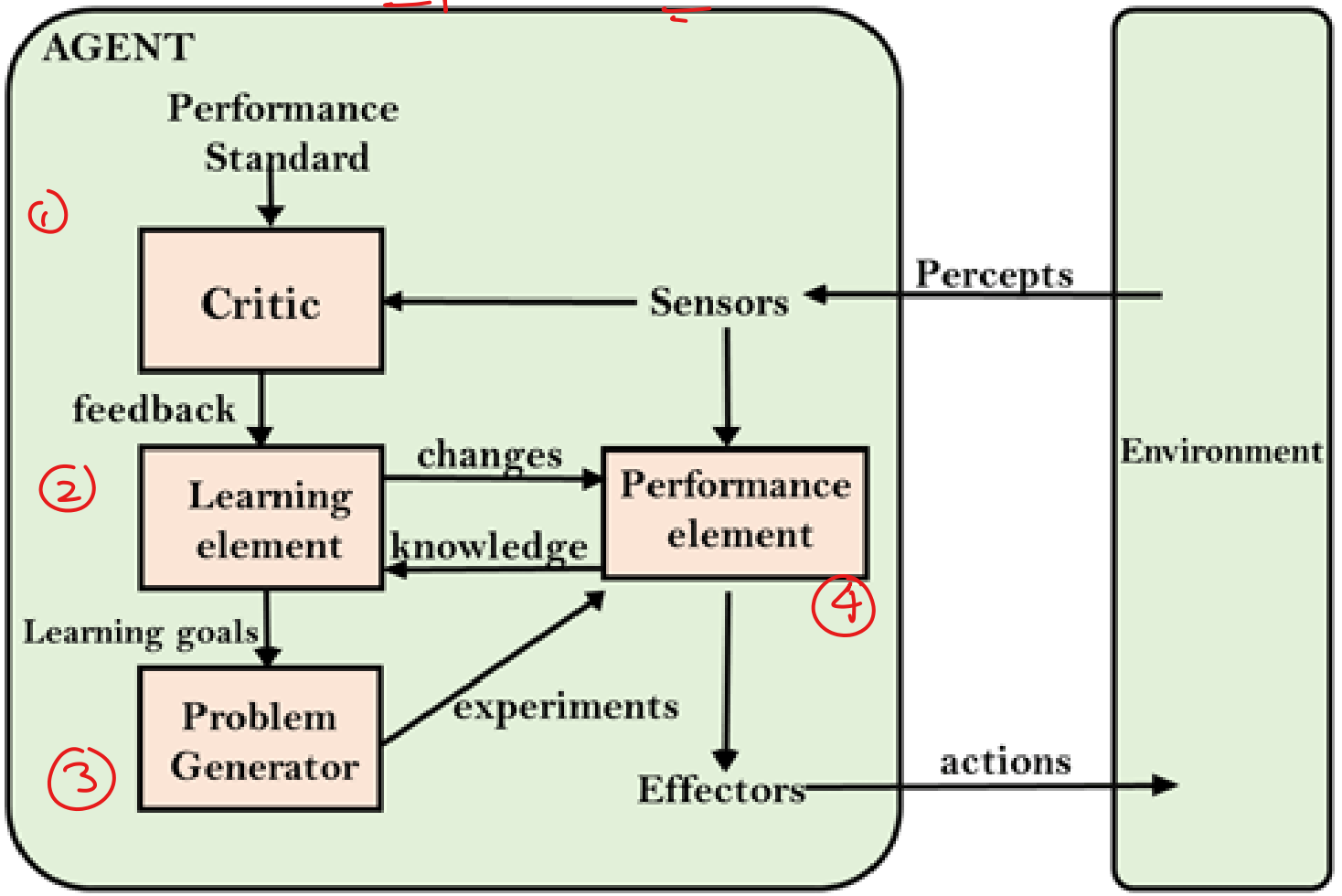
Flexible (Searching & Planning)

Example: Tour

PyB + definition

Utility based – happy & unhappy state

v imp. 2014 PYQ



15 A learning agent has mainly four conceptual components, which are:

(1) **Learning element :** It is responsible for making improvements by learning from the environment.

(2) **Critic:** Learning element takes feedback from critic which describes how well the agent is doing with respect to a fixed performance standard.

(3) **Performance element:** It is responsible for selecting external action.

(4) **Problem Generator:** This component is responsible for suggesting actions that will lead to new and informative experiences.

Knowledge Representation

1. Propositional Logic - PROLOG (Statements True / False)
2. FOPL – Quantifiers ∀ ∃
3. Rule Based – IF-Then-ELSE
   - Forward Chaining (Start -> Goal)
   - Backward Chaining (Goal -> Start)

   P48
4. Semantic Network – Graph (Nodes, Arcs)
5. Frames – Data Structure represent knowledge as Stereotype (Attributes (Fillers) or Slots (object))
6. Scripts – More refined than Frames (represent sequence of events)

Quick Revision of Planning Problem

Planning agents use look-ahead to find actions to contribute to goal achievement.

Planning agents differ from problem solvers in their use of more flexible representation of states, actions, goals, and plans.

The STRIPS language describes actions in terms of preconditions and effects.

Principle of least commitment is preferred.

Partial Order Planning is a sound and complete algorithm for planning using STRIPS representation.

Unit-10 Part-1

Blind Search (Small Search Space) learn it

2018

	BFS	DFS	Minimax OR Iterative Deepening (DFS+BFS)
Completeness	yes	NO	Yes
Time complexity	$O(b^{d+1})$	$O(b^m)$ linear	$O(b^d)$
Space	$O(b^{d+1})$	$O(bd)$	$O(bd)$
Optimal	Yes	NO	Yes

d = depth of shallowest soln
b = successor node (branching factor)

Problem: high Memory req.

d = m (max. depth loop)

Faster than BFS

Recursive approach

Informed (Real life Prob.)

Greedy Best Search	A*
No	Yes
$O(b^d)$	$O(b^d)$
$O(b^d)$	
No	Yes
Can stuck in loop (DFS)	high memory req.

$f(n) = g(n) + h(n)$

0: closest to Goal

1: fewest no. of steps

$h' = 0$, $g = 0$ Random

$g = 1$ BFS

Language Observe Types of Informed Uninformed

Heuristic

2019, 2018

1) Greedy Best First ⇒ $f(n) = h(n)$ [search for most promising nodes] [that seems to be closest to Goal node]

2) A* ⇒ $f(n) = g(n) + h(n)$ [search all possible path & then Selects Best One]

A* gives Optimal soln when heuristic is Admissible

2014

3) Admissible heuristic ⇒ Never Overestimates
or ↳ Example → Relate Real World.
optimistic

$h(n) \leq h^*(n)$
Estimated Cost LESS Actual cost

4) Monotonic (locally optimistic)
2018 follow Triangle Inequality property

$h(n_1) - h(n_2) \leq h^*(n_1) - h^*(n_2)$
estimated cost — actual cost

Uniform

$g(n) = \text{depth}$

Guaranteed to find Single soln at Minimum path Cost.

optimal & complete

UGC NET SET 2021

राष्ट्रीय परीक्षा एजेंसी
National Testing Agency
Excellence in Assessment

FREE MOCK TEST

CombineCS
The Extra Step

Computer Science

Price: ~~1000~~ free enroll

SUNDAY
5,12,19,26

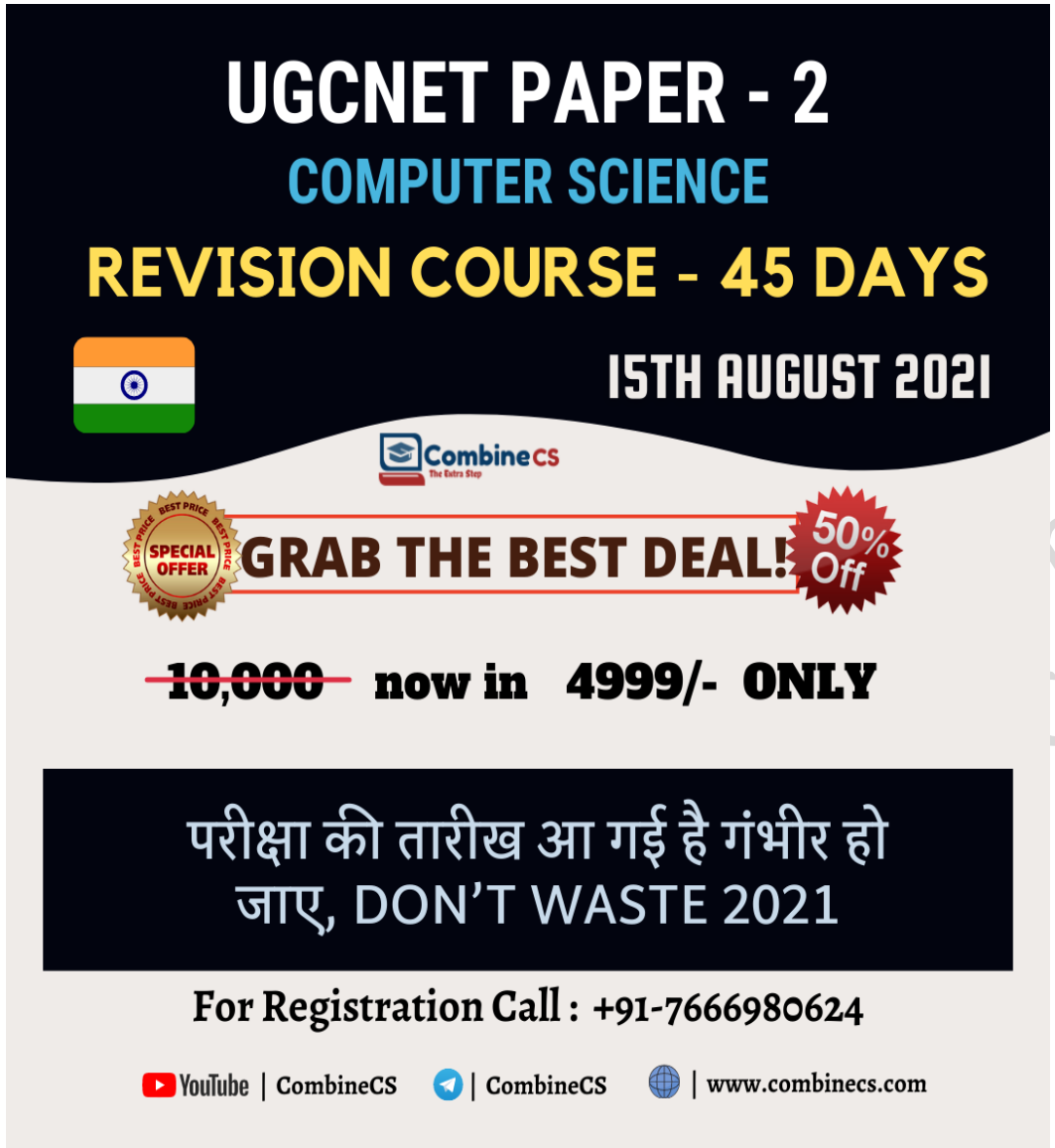
Time: 3:30pm

→ must enroll / register.

7669806124

CombineCS
The Extra Step

SUBSCRIBE



Enroll for Session 2022-23

👉 Follow us on Social media:

▶️ YouTube : https://www.youtube.com/c/CombineCSTheExtraStep
👥 Facebook : https://www.facebook.com/groups/combinecs
📸 Instagram : https://www.instagram.com/combinecs/
Telegram Group : https://t.me/RashmiCCS
Telegram Channel : https://t.me/combinecs

👉 Join our WhatsApp group for (NET/SET/GATE):
https://chat.whatsapp.com/GruovhRvste1nL8L2X1YQ3

👉 Join our WhatsApp group for (JOB Notifications) :
https://chat.whatsapp.com/ExM4CZ2ZKxzEgPvSfOXNFb

👉 Join our WhatsApp group for (Training & Placements):
https://chat.whatsapp.com/EB5umdja3BGJQijhxjEaij

For any query regarding notes, pdf, feedback, suggestions
Mail us: combinecs2020@gmail.com

🌟 For all our latest courses launched
visit: 🌍 combinecs.com
👨‍🎓 Student Support:- team.combinecs@gmail.com ☎️ 7666980624
==========

"Effort Never Dies"
👍 Like || Share || Comment || Subscribe


Thank you
Post your doubts in comment section.
Stay subscribed for all updates.