

後援：行基本変形サークル→フォローお待ちしております！ [@kit_matrix](#)



学科	
学生番号	
得点	

2019 年度

人工知能プログラミング本番レベル模試

ほぼ 2018 年度期末試験（記憶を頼りに再現）

（90 分）

★注意事項 解く前に絶対見てね！★

1. 問題用紙はこの冊子を含め全部で6ページである。
2. 解答用紙はなく、すべてこの冊子に示された解答欄に答えを記入すること。言い換えると、問題用紙は回収されるのでこの本番レベル模試は 2018 年度の人工知能プログラミングの問題を頭の中で覚えて再現したものである。なので、私ももやまの記憶力や日本語力が雑魚であっても炎上させないでほしい。
3. 本冊子に誤字・脱字・意味不明な日本語及び問題用紙の汚れ等があった場合、手を挙げてもどうにもできないので適当に想像しながら読むこと。
4. 問題は第1問～第3問があり、すべての問題が必答問題である。なのでわからない問題があった場合でもなんとかして解くこと。再テストや再履にならないように今がんばりましょう。あ、試験当日1限だから寝坊に注意！！これで再履とか嫌でしょ！
5. 解答には黒色鉛筆・シャープペンシル・ボールペン・万年筆・iPad・PC などなにを使っても構わないが、本番に筆記用具を忘れると悲惨な目に合うので試験当日は筆記用具を忘れないこと。
6. この本番レベル模試作成に関する私の給料は0なので悲しい限りです。また、来年度に備えて問題を記憶し、それを Word でタイピングしてくれる子を募集しています。
7. 頑張れば 100 点取れるテストです！ まじでまじで！

Prolog は解説ブログ作ったので見てね！ こちら→ <https://00m.in/cUnu9>

第1問

つぎの (1) ～ (10) のそれぞれを Prolog で実行した際に、実行に成功する場合は「Yes」、実行に失敗する場合は「No」、文法エラーになる場合は「x」を記入しなさい。また、実行に成功する、つまり「Yes」になる場合は実行後の各変数の値を答えなさい。ただし、実行に成功したあとも変数の値が定まらない場合は「不定」と記入すること。(配点 20)

- (1) $p([a,b,c,d]) = p([X|Y|Z])$.
- (2) $p([[a,b],c]) = p([X|Y])$.
- (3) $p([X,Y|Z]) = p([a,X,Y])$.
- (4) $p([a,b,U]) = p([X,b])$.
- (5) $[X,c] = [[a,b]|Y]$.
- (6) $[X|Y] = [a,b|U]$, $Y = [b,c,d]$.
- (7) $X = 2 + 3$.
- (8) $X \text{ is } 2 + 3$.
- (9) $X = (Y = 3)$.
- (10) $(X,Y) = (a,b,c)$.

解答欄

問題	Y/N/x	変数
(1)		
(2)		
(3)		
(4)		
(5)		
(6)		
(7)		
(8)		
(9)		
(10)		

第2問

下に示した Prolog プログラムの $p0(X,Y)$ を実行し、バックトラックしながら解を求めると、全部で何通りの解が得られるかを答えなさい。同様に $p1(X,Y), p2(X,Y), p3(X,Y), p4(X,Y), p5(X,Y)$ の実行について、バックトラックさせながら解を求めるとそれぞれ全部で何通りの解が得られるかを答えなさい。解答は下の解答欄に書くこと。解答欄以外に書いた答えは残念ながら読まれないので注意。(配点 20)

$p0(X,Y) :- \text{like}(X,Y), \text{animal}(Y).$ $p0(X,Y) :- \text{fly}(X).$ $p1(X,Y) :- !, \text{like}(X,Y), \text{animal}(Y).$ $p1(X,Y) :- \text{fly}(X).$ $p2(X,Y) :- \text{like}(X,Y), !, \text{animal}(Y).$ $p2(X,Y) :- \text{fly}(X).$ $p3(X,Y) :- \text{like}(X,Y), \text{animal}(Y), !.$ $p3(X,Y) :- \text{fly}(X).$ $p4(X,Y) :- \text{like}(X,Y), \text{animal}(Y).$ $p4(X,Y) :- !, \text{fly}(X).$ $p5(X,Y) :- \text{like}(X,Y), \text{animal}(Y).$ $p5(X,Y) :- \text{fly}(X), !.$	$\text{like}(\text{bone}, \text{sora}).$ $\text{like}(\text{catfood}, \text{sora}).$ $\text{like}(\text{meat}, \text{mike}).$ $\text{like}(\text{fish}, \text{mike}).$ $\text{like}(\text{cheese}, \text{mike}).$ $\text{like}(\text{nekojarashi}, \text{leo}).$ $\text{animal}(\text{sora}).$ $\text{animal}(\text{mike}).$ $\text{fly}(\text{chocobo}).$ $\text{fly}(\text{bird}).$
---	--

解答欄

$p0(X,Y)$	$p1(X,Y)$	$p2(X,Y)$	$p3(X,Y)$	$p4(X,Y)$	$p5(X,Y)$
通り	通り	通り	通り	通り	通り

第3問

ある空港に運ばれてきた果物（果物以外がないとは言っていない）の名前とその量を指し示すリスト DATA（下に示す）があるとする。

このリストには、[hnd,40,apple] のように、第1引数に空港コード（3桁レター、気になる人は調べてください）、第2引数にその量がトン単位で、第3引数に果物（果物以外がないとは言っていない）というリストの形を利用して作成する。

```
DATA = [  
[hnd,40,apple],      [itm,90,apple],      [ngo,55,grape],  
[cts,75,melon],      [hnd,75,orange],      [fuk,100,ramen],  
[hij,25,lemon],      [oka,25,mango],      [hnd,60,apple],  
[kmi,100,banana],    [nrt,20,rakkasei], [oka,30,banana],  
[cts,10,watermelon], [isg,10,salt],      [nrt,40,cucumber],  
[hnd,200,strawberry],[itm,45,orange],  
[fuk,40,sumomo],     [oka,20,pineapple] ].
```

※実行例を示すときにこのリストを書くとスペースの無駄極まりないので、このリストを DATA と呼ぶことにする。

このリスト DATA に対して、以下の(1)～(3)のプログラムを指示に従って作成をなさい。
ただし、(1)～(3)すべてにおいて、下請けプログラムは作成しないこと。

（配点 60）

(1)

ある空港に運ばれてきた**すべての果物の量**を計算するプログラム `ap_sum` を作成したい。
第 1 引数にリスト `DATA` を与え、第 2 引数で指定した空港 `Loc` からすべての果物の運ばれてきた量の合計を第 3 引数 `Sum` に求めるプログラム `ap_sum(DATA, Loc, Sum)` を作成しなさい。バックトラックにより別の解が出ないように注意すること。

実行結果例 1 : `ap_sum(DATA, itm, Ans) → Ans = 135.`

実行結果例 2 : `ap_sum(DATA, hnd, Ans) → Ans = 375.`

(1)の解答欄

(2)

果物を指定すると、どの空港から運ばれたかを表示するプログラム `fruit_log` を作成したい。

第 1 引数にリスト `DATA` を与え、第 2 引数で指定した果物 `Fruit` から運ばれてきた空港の一覧をリストで表示する第 3 引数 `Carry` に求めるプログラム `fruit_log(DATA, Fruit, Carry)` を作成しなさい。ただし、**同じ空港は 2 回以上重複して出さないようにすること**。また、バックトラックにより別の解が出ないように注意すること。

実行結果例 1 : `fruit_loc(DATA, banana, Carry) → Carry = [kmi, oka].`

実行結果例 2 : `fruit_loc(DATA, apple, Carry) → Carry = [itm, hnd].`

(2)の解答欄

(3)

空港を指定すると、指定した空港から運ばれた果物の量と名前を表示するプログラム `ap_info` を作成したい。

第1引数にリスト `DATA` を与え、第2引数で指定した空港 `Loc` から運ばれてきた果物の量と名前を表示する第3引数 `Info` に求めるプログラム `ap_info(DATA, Loc, Ans)` を作成しなさい。

ただし、解 `Info` は第1引数を量、第2引数を果物名とした要素数2のリストとし、バックトラックすることに要素を1つずつ順番に求めるものとする。

実行結果例1: `ap_info(DATA, fuk, Ans)` .

`Ans = [100, ramen] ? ;`

`Ans = [40, sumomo] ? ;`

`no`

実行結果例2: `ap_info(DATA, hnd, Ans)` .

`Ans = [40, apple] ? ;`

`Ans = [75, orange] ? ;`

`Ans = [60, apple] ? ;`

`Ans = [20, strawberry] ? ;`

`no`

(3)の解答欄

問題は以上です。おつかれさまでした！

期末試験もファイト！ 君ならできる！