$EPS){ $ff = 1; } } if (($ff == 0) || ($i > $MAX\_COUNT)) { break; } } return(($lambda - $n) / ($n - 1)); } /\* --------------------------------------------------------- AHP\_Weight\_Calc 一対比較行列から，重要度，整合度を求める． $Pc\_Value[]を，一対比較行列に変換して，関数AHP\_WCを呼び出す． 入力： $n: 一対比較の項目数 $Pc\_Value[] : 一対比較値 一対比較の順番にしたがって，順番に値が入っている． 1/a は，-a で表現する $Pc\_Value[0] <- 1番目の項目 ： 2番目の項目 : : : $Pc\_Value[$n-1] <- 1番目の項目 ： n番目の項目 : : : $Pc\_Value[($n\*$n-1)2] <- n-1番目の項目 ： n番目の項目 $b[] : 重要度が計算されセットされる(合計が1に正規化されている)． 呼び出すときは，&$b のようにする． 戻り値： 整合度(C.I.) ---------------------------------------------------------- \*/ function AHP\_Weight\_Calc($n,$Pc\_Value,&$bb){ $point = 0; for ($i=0; $i < $n ; $i++){ for ($j=0; $j < $n ; $j++){ if ( $j > $i ){ $t = floatval($Pc\_Value[$point]); if ($t < 0 ){ $a[$i][$j] = 1 / (-1 \* $t); } else { $a[$i][$j] = $t; } $point++; } if ($j == $i ){ $a[$i][$j] = 1; } if ($j < $i ){ $a[$i][$j] = 1 / $a[$j][$i]; } } } return( AHP\_WC($n, $a, $bb)); } /\* --------------------------------------------------------- AHP\_GV\_Calc 総合評価値を計算する関数 入力： $n: 評価項目の数 $m: 代替案の数 $kobetu\_h[][]: 個別評価値 $kobetu\_h[$i][$j] : $i-1 番目の基準についての$j-1番目の代替案の個別評価値 例：$kobetu\_h[1][0] : 2番目の基準の1番目の代替案の個別評価値 $GV\_value[] : 総合評価値 $GV\_Rank[] : $GV\_Rank[$i] :$i-1 位の代替案の番号 $GV\_Rank[0] -> 2：1位の代替案の番号は，2で（3番目の）代替案 $cc[][] : 重要度をかけた後の個別評価値 戻り値： なし ---------------------------------------------------------- \*/ // 総合評価値を計算 function AHP\_GV\_Calc($n,$m,$bb,$kobetu\_h,&$GV\_value, &$GV\_Rank, &$cc){ for ($i = 0 ; $i < $m ; $i++ ){ // 代替案分繰り返す $GV\_value[$i] = 0; for($j = 0 ; $j < $n ; $j++ ){ $cc[$j][$i] = $bb[$j] \* $kobetu\_h[$j][$i]; $GV\_value[$i] = $GV\_value[$i] + $cc[$j][$i]; } } // ソートして順位を求める for ($i = 0 ; $i < $m ; $i++ ){ // 代替案分繰り返す $GV\_Rank[$i] = $i; $rr\_GV[$i] = $GV\_value[$i]; } $ff=1; while ($ff==1){ $ff=0; for($i = 0 ; $i < ($m-1) ; $i++ ){ if ( $rr\_GV[$i] < $rr\_GV[$i+1] ){ $ff=1; $t =$rr\_GV[$i]; $rr\_GV[$i] = $rr\_GV[$i+1]; $rr\_GV[$i+1] = $t; $t =$GV\_Rank[$i]; $GV\_Rank[$i] = $GV\_Rank[$i+1]; $GV\_Rank[$i+1] = $t; } } } } /\* ---------------------------------------------------- AHPのアンケート用紙を作成する． $nn : 一対比較の項目数 ($nn \* ($nn -1) /2) 回一対比較が行われる $Evi\_Name[] : 項目名 配列．1から$nn 結果を入れる名前，PC1 ～ PC($nn \* ($nn -1) /2) ----------------------------------- \*/ function AHP\_enqtable($nn, $Evi\_Name){ $mm = ($nn \* ($nn -1) /2); // 一対比較の回数 ?> = 1; $ik-- ){ ?> = -9; $ik-- ){ ?> = $nn ){ $il++; $ir = $il+1; } } ?>

|  | 左  の  項  目  が  圧  倒  的  に  重  要 |  | 左  の  項  目  が  う  ん  と  重  要 |  | 左  の  項  目  が  か  な  り  重  要 |  | 左  の  項  目  が  少  し  重  要 |  | 左  右  同  じ  く  ら  い  重  要 |  | 右  の  項  目  が  少  し  重  要 |  | 右  の  項  目  が  か  な  り  重  要 |  | 右  の  項  目  が  う  ん  と  重  要 |  | 右  の  項  目  が  圧  倒  的  に  重  要 |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |

<?php }