

程式設計作業一 利率因子計算機

B09701203 白敘廷

一、動機

由於就讀管理相關科系，因此在近期會計單元上，接觸利息與年金相關計算。其中，在查詢複利現值表、複利終值表、年金現值表、年金終值表的利率因子時，許多人常因為表格數字緊密複雜，而導致計算和操作上的錯誤。因此，此次程式作業將針對四張表格進行操作，運用 C# 中的流程控制、迴圈、字串格式化等技巧實作表格的產生，並達到學以致用的效果。

複利終值表										
N	1.00%	2.00%	3.00%	4.00%	5.00%	6.00%	7.00%	8.00%	9.00%	10.00%
1	1.010	1.020	1.030	1.040	1.050	1.060	1.070	1.080	1.090	1.100
2	1.020	1.040	1.061	1.082	1.103	1.124	1.145	1.166	1.188	1.210
3	1.030	1.061	1.093	1.125	1.158	1.191	1.225	1.260	1.295	1.331
4	1.041	1.082	1.126	1.170	1.216	1.262	1.311	1.360	1.412	1.464
5	1.051	1.104	1.159	1.217	1.276	1.338	1.403	1.469	1.539	1.611
6	1.062	1.126	1.194	1.265	1.340	1.419	1.501	1.587	1.677	1.772
7	1.072	1.149	1.230	1.316	1.407	1.504	1.606	1.714	1.828	1.949
8	1.083	1.172	1.267	1.369	1.477	1.594	1.718	1.851	1.993	2.144
9	1.094	1.195	1.305	1.423	1.551	1.689	1.838	1.999	2.172	2.358
10	1.105	1.219	1.344	1.480	1.629	1.791	1.967	2.159	2.367	2.594
11	1.116	1.243	1.384	1.539	1.710	1.898	2.105	2.332	2.580	2.853
12	1.127	1.268	1.426	1.601	1.796	2.012	2.252	2.518	2.813	3.138
13	1.138	1.294	1.469	1.665	1.886	2.133	2.410	2.720	3.066	3.452
14	1.149	1.319	1.513	1.732	1.980	2.261	2.579	2.937	3.342	3.797
15	1.161	1.346	1.558	1.801	2.079	2.397	2.759	3.172	3.642	4.177
16	1.173	1.373	1.605	1.873	2.183	2.540	2.952	3.426	3.970	4.595
17	1.184	1.400	1.653	1.948	2.292	2.693	3.159	3.700	4.328	5.054
18	1.196	1.428	1.702	2.026	2.407	2.854	3.380	3.996	4.717	5.560
19	1.208	1.457	1.754	2.107	2.527	3.026	3.617	4.316	5.142	6.116
20	1.220	1.486	1.806	2.191	2.653	3.207	3.870	4.661	5.604	6.727
21	1.232	1.516	1.860	2.279	2.786	3.400	4.141	5.034	6.109	7.400
22	1.245	1.546	1.916	2.370	2.925	3.604	4.430	5.437	6.659	8.140
23	1.257	1.577	1.974	2.465	3.072	3.820	4.741	5.871	7.258	8.954
24	1.270	1.608	2.033	2.563	3.225	4.049	5.072	6.341	7.911	9.850
25	1.282	1.641	2.094	2.666	3.386	4.292	5.427	6.848	8.623	10.835

附圖：複利終值表

二、構想解說

利率因子有固定公式，具有可計算、可預測性的特質；再者，利率因子由兩個變數-期數(n)、利率(rate)所控制。

項目	公式
複利現值表	<code>Math.Pow((1+rate),-1*n)</code>
複利終值表	<code>Math.Pow((1+rate),n)</code>
年金現值表	<code>(1-Math.Pow((1+rate),-1*n))/rate</code>
年金終值表	<code>(Math.Pow((1+rate),n)-1)/rate</code>

綜合上述兩點推測程式適合以 for 迴圈進行實作。而此程式有兩大功能：

1. 查詢現值/複利因子

使用者可輸入利率與期數，查詢相對應的年金現值因子、年金終值因子、複利現值因子與複利終值因子。

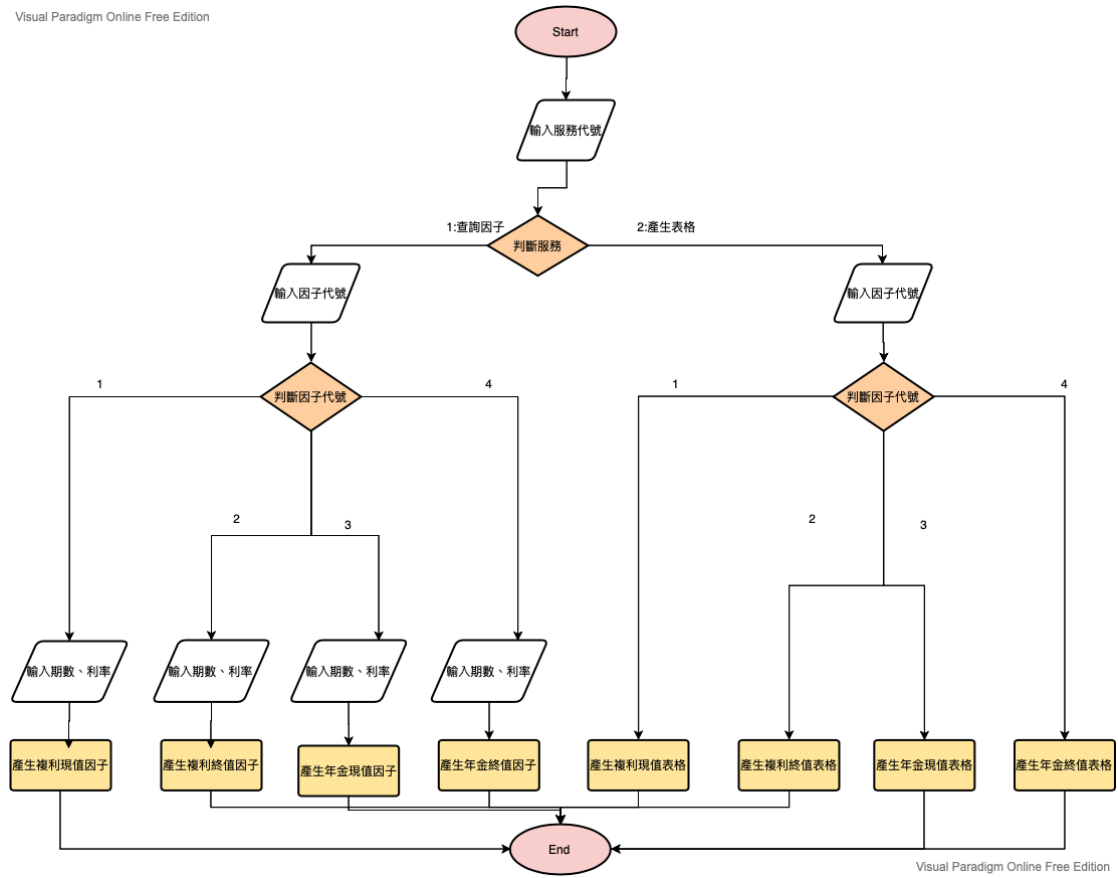
2. 產生表格

使用者可利用程式，自動產生[10X25]的年金現值表、年金終值表、複利現值表、複利終值表。

最後，此程式旨於解決因表格數字緊密複雜而導致的計算錯誤，因此本程式將更加著重於資訊的呈現方式，盡能達到使用者友善的效果。

三、流程圖

Visual Paradigm Online Free Edition




```

        Console.WriteLine("年金現值為:{0}", ((1 - Math.Pow((1 + rate), -
1 * n)) / rate).ToString("f3"));

        break;

        case 4:

            Console.WriteLine("年金終值為:{0}", ((Math.Pow((1 + rate), n) -
1) / rate).ToString("f3"));

            break;

        default:

            break;

    }

}

else if (input1 == 2)

{

    Console.Write("請選擇欲產生之表格\n1:[複利現值表] 2:[複利終值表] 3:[年金現值表] 4:[年金
終值表]\n>>>");

    int input2 = int.Parse(Console.ReadLine());

    Console.WriteLine();

    switch(input2)

    {

        case 1:

            Console.WriteLine("\n*複利現值表*\n");

            break;

        case 2:

            Console.WriteLine("\n*複利終值表*\n");

            break;

        case 3:

            Console.WriteLine("\n*年金現值表*\n");

            break;

        case 4:

            Console.WriteLine("\n*年金終值表*\n");

            break;

    }

}

for(int n = 0; n <= 25; n++)

{

    string data;

    data = "";

```

```

for(int i = 0; i<=10; i++)
{
    if(i == 0 && n == 0)
    {
        data += String.Format("{0,2}", "期數/利率");
    }
    else if(n == 0)
    {
        data += String.Format("{0,7}%", i);
    }
    else
    {
        if(i == 0)
        {
            data += String.Format("{0,8}", n);
        }
        else
        {
            double rate;
            rate = 0.01*i;
            double CurrentValue;
            CurrentValue = 0;
            switch(input2)
            {
                case 1:
                    CurrentValue = Math.Pow((1+rate), -1*n);
                    break;
                case 2:
                    CurrentValue = Math.Pow((1+rate), n);
                    break;
                case 3:
                    CurrentValue = (1-Math.Pow((1+rate), -1*n))/rate;
                    break;
                case 4:
                    CurrentValue = (Math.Pow((1+rate), n)-1)/rate;
                    break;
            }
            string str_CurrentValue;
            str_CurrentValue = "";
            str_CurrentValue = CurrentValue.ToString("f3"); // 取小數點三位

```

```
        data += String.Format("{0,8}",str_CurrentValue);  
    }  
}  
  
}  
Console.WriteLine(data);  
}  
  
}  
  
}  
}
```

五、程式測試執行結果

1. 產生利率因子

```
[tingting@baixutingdeMacBook-Air Project2 % dotnet run
利息表產生器

請選擇服務項目
1:[查詢] 2:[產生表格]: 2
[>>>1

請選擇欲查詢之表格
1:[複利現值表] 2:[複利終值表] 3:[年金現值表] 4:[年金終值表]
[>>>3

請輸入期數(整數): 4
請輸入利率(%): 6
年金現值為:3.465
```

2. 產生表格

利息表產生器

請選擇服務項目

1:[查詢] 2:[產生表格]: 2

[>>>2

請選擇欲產生之表格

1:[複利現值表] 2:[複利終值表] 3:[年金現值表] 4:[年金終值表]

[>>>3

年金現值表

期數/利率	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
1	0.990	0.980	0.971	0.962	0.952	0.943	0.935	0.926	0.917	0.909
2	1.970	1.942	1.913	1.886	1.859	1.833	1.808	1.783	1.759	1.736
3	2.941	2.884	2.829	2.775	2.723	2.673	2.624	2.577	2.531	2.487
4	3.902	3.808	3.717	3.630	3.546	3.465	3.387	3.312	3.240	3.170
5	4.853	4.713	4.580	4.452	4.329	4.212	4.100	3.993	3.890	3.791
6	5.795	5.601	5.417	5.242	5.076	4.917	4.767	4.623	4.486	4.355
7	6.728	6.472	6.230	6.002	5.786	5.582	5.389	5.206	5.033	4.868
8	7.652	7.325	7.020	6.733	6.463	6.210	5.971	5.747	5.535	5.335
9	8.566	8.162	7.786	7.435	7.108	6.802	6.515	6.247	5.995	5.759
10	9.471	8.983	8.530	8.111	7.722	7.360	7.024	6.710	6.418	6.145
11	10.368	9.787	9.253	8.760	8.306	7.887	7.499	7.139	6.805	6.495
12	11.255	10.575	9.954	9.385	8.863	8.384	7.943	7.536	7.161	6.814
13	12.134	11.348	10.635	9.986	9.394	8.853	8.358	7.904	7.487	7.103
14	13.004	12.106	11.296	10.563	9.899	9.295	8.745	8.244	7.786	7.367
15	13.865	12.849	11.938	11.118	10.380	9.712	9.108	8.559	8.061	7.606
16	14.718	13.578	12.561	11.652	10.838	10.106	9.447	8.851	8.313	7.824
17	15.562	14.292	13.166	12.166	11.274	10.477	9.763	9.122	8.544	8.022
18	16.398	14.992	13.754	12.659	11.690	10.828	10.059	9.372	8.756	8.201
19	17.226	15.678	14.324	13.134	12.085	11.158	10.336	9.604	8.950	8.365
20	18.046	16.351	14.877	13.590	12.462	11.470	10.594	9.818	9.129	8.514
21	18.857	17.011	15.415	14.029	12.821	11.764	10.836	10.017	9.292	8.649
22	19.660	17.658	15.937	14.451	13.163	12.042	11.061	10.201	9.442	8.772
23	20.456	18.292	16.444	14.857	13.489	12.303	11.272	10.371	9.580	8.883
24	21.243	18.914	16.936	15.247	13.799	12.550	11.469	10.529	9.707	8.985
25	22.023	19.523	17.413	15.622	14.094	12.783	11.654	10.675	9.823	9.077

六、參考文獻

String.Format 方法

<https://docs.microsoft.com/zh-tw/dotnet/api/system.string.format?view=net-5.0>

年金現值終值表

<http://w3.uch.edu.tw/shopping/homework/年金現值終值表.pdf>