```
> #The This code calculates the total reliability of different
    systems with increasing redundancy by using reliability
    values for each component. It shows how adding more
    redundant components improves the overall reliability.
  #As you see I define the Reliability formula for 1 M,
    TMR, 5 MR and 7 MR as a function

  # Define reliability of a single module as Rm
  # r will represent the reliability of a single module
  Rm := r;

   # 1 M reliability formula
   R_1M := Rm -> Rm;

  # TMR reliability formula
  R_TMR := Rm -> Rm^3 + 3 * Rm^2 * (1 K  Rm);

  # 5 MR reliability formula
  R_5MR := Rm -> Rm^5 + 5 * Rm^4 * (1 K  Rm) + 10 * Rm^3 * (1 K  Rm)^2;

  # 7 MR reliability formula
  R_7MR := Rm -> Rm^7 + 7 * Rm^6 * (1 K  Rm) + 21 * Rm^5 * (1 K  Rm)^2 + 35 * Rm^4
       * (1 K  Rm)^3;
```

$$Rm := r$$

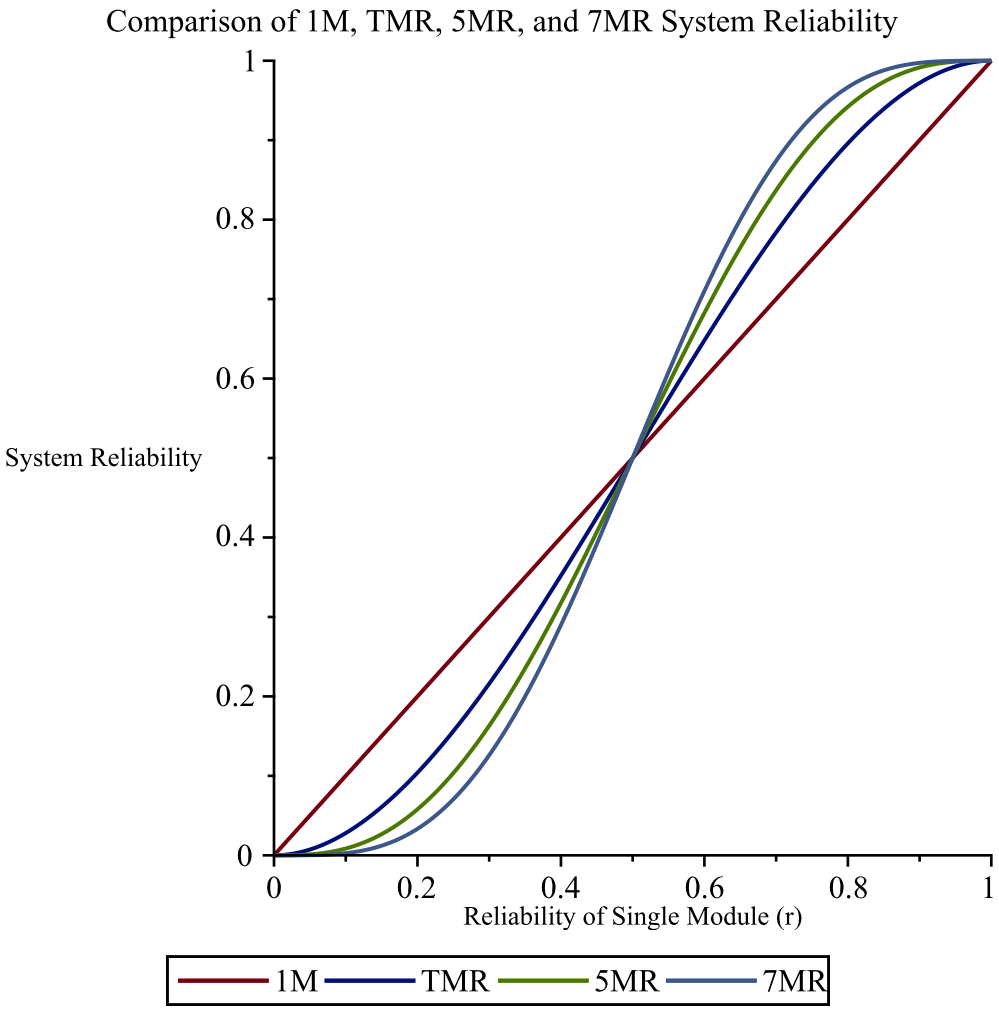
$$R\_1M := Rm \mapsto Rm$$

$$R\_TMR := Rm \mapsto Rm^3 + 3 \cdot Rm^2 \cdot (1\,\mathrm{K}\ \ Rm)$$

$$R\_5MR := Rm \mapsto Rm^5 + 5 \cdot Rm^4 \cdot (1\,\mathrm{K}\ \ Rm) + 10 \cdot Rm^3 \cdot (1\,\mathrm{K}\ \ Rm)^2$$

$$R\_7MR := Rm \mapsto Rm^7 + 7 \cdot Rm^6 \cdot (1\,\mathrm{K}\ \ Rm) + 21 \cdot Rm^5 \cdot (1\,\mathrm{K}\ \ Rm)^2 + 35 \cdot Rm^4 \cdot (1\,\mathrm{K}\ \ Rm)^3 \qquad \textbf{(1)}$$

```
> # Then I will plot the formulas in the range from 0 to 1 for component reliability of different systems (1M, TMR, 5MR, and 7MR).
# Plot the reliability functions
plot( [R_1M(r), R_TMR(r), R_5MR(r), R_7MR(r) ], r=0 .. 1,
    labels= ["Reliability of Single Module (r)", "System Reliability"],
    legend= ["1M", "TMR", "5MR", "7MR"],
    title="Comparison of 1M, TMR, 5MR, and 7MR System Reliability");
```



As you see by incresing the number of components with R > 0.5 the system reliability will increase also but its opposite for R < 0.5

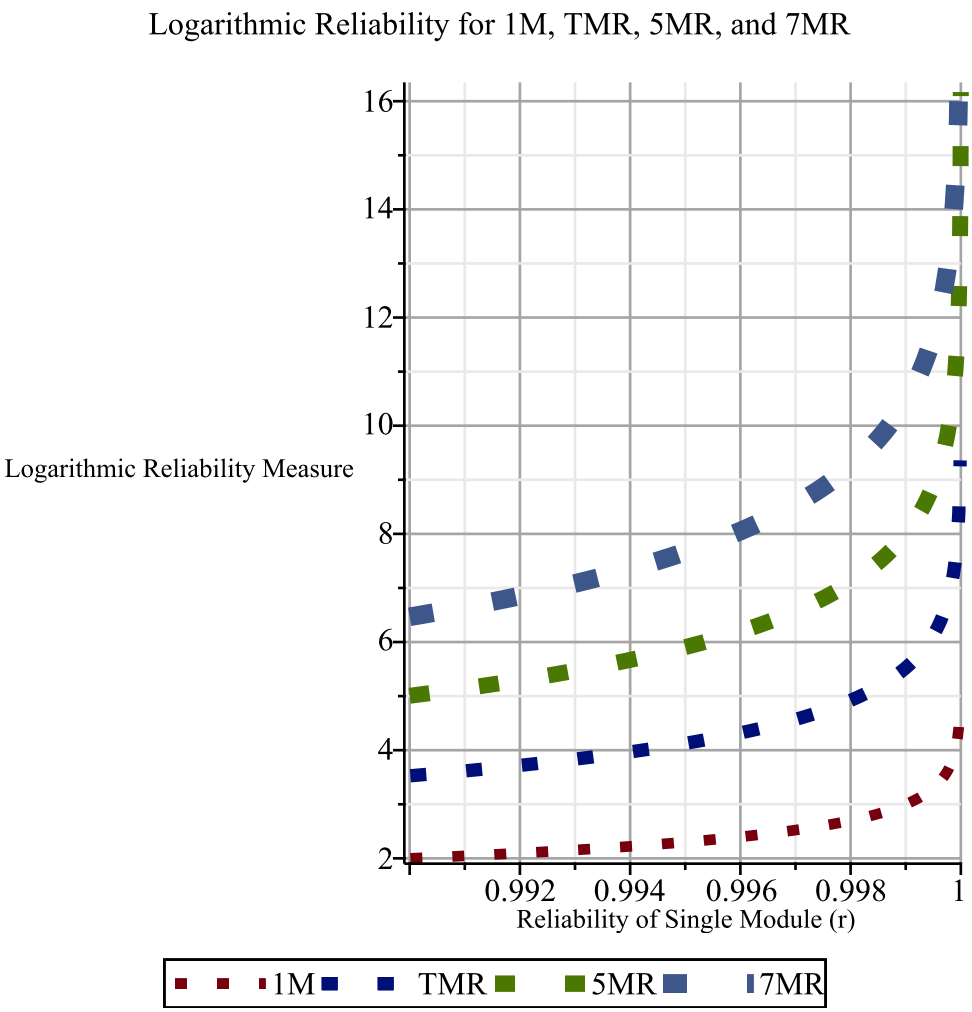
```
> # I have also plotted their logarithmic graphs in the range
    of 0.99 to 1 for better visualization, to show their
    growth rates.
  plot_log_reliability := proc( )
    local log_TMR, log_5MR, log_7MR, log_1M;

    # Logarithmic reliability measures
    log_1M := r→ K log10(1 K R_1M(r) ); # Logarithmic reliability for one module
    log_TMR := r→ K log10(1 K R_TMR(r) );
    log_5MR := r→ K log10(1 K R_5MR(r) );
    log_7MR := r→ K log10(1 K R_7MR(r) );

    # Plotting the results
    plot( [ log_1M(r), log_TMR(r), log_5MR(r), log_7MR(r) ],
      r= 0.99 .. 1,
      labels= [ "Reliability of Single Module (r)", "Logarithmic Reliability Measure" ],
      legend= [ "1M", "TMR", "5MR", "7MR" ],
      thickness= [ 4, 5, 6, 7 ],
      linestyle= [ dot, dot, dot, dot ],
      gridlines= true,
      title= "Logarithmic Reliability for 1M, TMR, 5MR, and 7MR"
    );
  end proc;

  # Call the procedure to plot
  plot_log_reliability( );


plot_log_reliability := proc( )
  local log_TMR, log_5MR, log_7MR, log_1M;
  log_1M := r↦K log10(1 − R_1M(r) );
  log_TMR := r↦K log10(1 − R_TMR(r) );
  log_5MR := r↦K log10(1 − R_5MR(r) );
  log_7MR := r↦K log10(1 − R_7MR(r) );
  plot( [ log_1M(r), log_TMR(r), log_5MR(r), log_7MR(r) ], r= 0.99 ..1, labels
   = [ "Reliability of Single Module (r)", "Logarithmic Reliability Measure" ], legend= [ "1M",
  "TMR", "5MR", "7MR" ], thickness= [ 4, 5, 6, 7 ], linestyle= [ dot, dot, dot, dot ], gridlines= true,
  title= "Logarithmic Reliability for 1M, TMR, 5MR, and 7MR" )
end proc
```


Logarithmic Reliability for 1M, TMR, 5MR, and 7MR
Logarithmic Reliability Measure
16
14
12
10
8
6
4
2
0.992
0.994
0.996
0.998
1
Reliability of Single Module (r)
1M
TMR
5MR
7MR

```
> # In the below part I'll answer the second part of the
    question, I have dealt with drawing a table for the
    values 0.999, 0.99999, and 0.9999999 for R. Using a for
    loop, I call the functions I had defined earlier each time
    and print their output.
  # Define r values and function labels
  r_values := [0.999, 0.99999, 0.9999999];
  reliability_functions := [R_1M, R_TMR, R_5MR, R_7MR];
  function_names := ["R_1M ", "R_TMR", "R_5MR", "R_7MR"];

  # Print header
  printf("\n\nReliability Values for Different Systems:\n\n");
  printf("\t\tr=0.999\t\tr=0.99999\t\tr=0.9999999\n");
      printf("----------------------------------------------------------------------------------------------------
      \n");

  # Loop through each reliability function and print values for each r
  for i from 1 to 4 do
    printf("%s\t", function_names[i]);
    for r in r_values do
      printf("|\t%.10f", evalf(reliability_functions[i](r)));
    end do;
    printf("\n");
  end do;

      printf("----------------------------------------------------------------------------------------------------
      \n");
```

$$r\_values := [0.999, 0.99999, 0.9999999]$$

$$reliability\_functions := [R\_1M, R\_TMR, R\_5MR, R\_7MR]$$

$$function\_names := [\text{"R\_1M "}, \text{"R\_TMR"}, \text{"R\_5MR"}, \text{"R\_7MR"}]$$

Reliability Values for Different Systems:

	r=0.999	r=0.99999	r=0.9999999
R_1M	0.9990000000	0.9999900000	0.9999999000
R_TMR	0.9999970020	0.9999999997	1.0000000000
R_5MR	0.9999999900	1.0000000000	1.0000000000
R_7MR	1.0000000000	1.0000000000	1.0000000000

```
>
```