<u>Musterlösung:</u> AlgoDat (Magister) **1. Serie**

1.) 3 Pkt. Obere, Untere und Exakte Schranke

$$g(n) = \frac{1}{10}n^5 + \frac{9}{10}n^3 - n = n^5(\frac{1}{10} + \frac{9}{10n^2} - \frac{1}{n^4}) \le n^5(\frac{1}{10} + \frac{9}{10{n_0}^2})$$

$$\Rightarrow n_0 = 1$$

$$g(n) \le n^5(\frac{1}{10} + \frac{9}{10}) = n^5 \times c$$

$$c = (\frac{1}{10} + \frac{9}{10}) = 1$$

$$g(n) \in O(n^5)$$

$$g(n) = n^5(\frac{1}{10} + \frac{9}{10n^2} - \frac{1}{n^4}) \ge n^5(\frac{1}{10} - \frac{1}{{n_0}^4})$$

$$\Rightarrow n_0 = 2$$

$$g(n) \ge n^5(\frac{1}{10} - \frac{1}{16}) = n^5 \times c$$

$$c = (\frac{1}{10} - \frac{1}{16}) = \frac{3}{80}$$

$$g(n) \in \Omega(n^5)$$

$$\Rightarrow g(n) \in \Theta(n^5)$$

2.) 8 Pkt. (je 2 Pkt.) Aussage und Begründung

a) $n \times \log_2(n) \in O(n^2)$ -> Ja, da $\log_2(n) \le n \Rightarrow n \times \log_2(n) \le n^2 \Rightarrow O(n^2)$ ab n=3

b) $n \in O(\log_2(n^2))$ -> Nein, da $\log_2(n^2) = 2 \times \log(n) \le n$ ab n=4

c) $n + \log_2(n) \in \Theta(n)$ -> Ja, da $n + \log_2(n) \le n + n = 2n = c \times n$ mit $c = 2$ und
$n + \log_2(n) \ge n = c \times n$ mit $c = 1$, so $(n + \log_2(n)) \in O(n) \wedge (n + \log_2(n)) \in \Omega(n)$

d) $n + \sqrt{n} \in \Theta(n)$ -> Ja, da $n + \sqrt{n} = n(1 + \frac{1}{\sqrt{n}}) \le n(1+1) = 2n = c \times n$ mit $c = 2$ und
$n + \sqrt{n} = n(1 + \frac{1}{\sqrt{n}}) \ge n \times 1 = n = c \times n$ mit $c = 1$

3.) 10 Pkt. (je 2 Pkt.) Obere, Untere und Exakte Schranke mit c,n_0:

a) $g(n) = 3n^2 + 4n = n^2(3 + \frac{4}{n}) \le n^2(3 + \frac{4}{n_0}) \quad \forall n_0 \ge 1 \wedge n \ge n_0$

$n^2(3 + \frac{4}{n_0}) = n^2(3+4) = c \times n^2$ mit $n_0 = 1$ und $c = 7$

$g(n) = 3n^2 + 4n = n^2(3 + \frac{4}{n}) \ge n^2 \times 3 \qquad \forall n_0 \ge 1$ und $c = 3$

$g(n) \in O(n^2), g(n) \in \Omega(n^2), g(n) \in \Theta(n^2)$

b) $g(n) = 2^{n+10} = 2^n \times 2^{10} \le c \times 2^n \qquad \forall n_0 \ge 1$ und $c = 1024$

$g(n) = 2^{n+10} = 2^n \times 2^{10} \ge c \times 2^n \qquad \forall n_0 \ge 1$ und $c = 1024$

$g(n) \in O(2^n), g(n) \in \Omega(2^n), g(n) \in \Theta(2^n)$

c) $g(n) = n\sqrt{n} = n^{\frac{3}{2}} \leq c \times n^k \quad \forall n_0 \geq 1$ und $c = 1$ und $k = \frac{3}{2}$

$g(n) = n\sqrt{n} = n^{\frac{3}{2}} \geq c \times n^k \quad \forall n_0 \geq 1$ und $c = 1$ und $k = \frac{3}{2}$

$g(n) \in O(n^k), g(n) \in \Omega(n^k), g(n) \in \Theta(n^k)$

d) $g(n) = n\sqrt{n} + n^2 = n^2(1 + \frac{1}{\sqrt{n}}) \leq n^2(1 + \frac{1}{\sqrt{n_0}}) \quad \forall n_0 \geq 1 \wedge n \geq n_0$

$n_0 = 1$

$n^2(1 + \frac{1}{\sqrt{n_0}}) = 2n^2 = c \times n^2$ mit $c = 2$

$g(n) = n\sqrt{n} + n^2 = n^2(1 + \frac{1}{\sqrt{n}}) \geq n^2 \times 1 = c \times n^2 \quad \forall n_0 \geq 1 \wedge n \geq n_0$

mit $n_0 = 1$ und $c = 1$

$g(n) \in O(n^2), g(n) \in \Omega(n^2), g(n) \in \Theta(n^2)$

e) $g(n) = 2^{n+10} + \log(n^2) = 2^n \times (2^{10} + \frac{\log(n)}{2^{n-1}}) \leq 2^n \times (2^{10} + \frac{\log(n_0)}{2^{n_0-1}})$

$\forall n_0 \geq 2 \wedge n \geq n_0$ mit $n_0 = 2$ folgt

$2^n(2^{10} + \frac{\log(n_0)}{2^{n_0-1}}) = 2^n(2^{10} + 0{,}5) = c \times 2^n$ mit $c = 1024{,}5 \approx 1025$

$g(n) = 2^{n+10} + \log(n^2) = 2^n \times (2^{10} + \frac{\log(n)}{2^{n-1}}) \geq 2^n \times 1024 = c \times 2^n$

$\forall n_0 \geq 2 \wedge n \geq n_0$ und $c = 1024$

$g(n) \in O(2^n), g(n) \in \Omega(2^n), g(n) \in \Theta(2^n)$

4.) 6Pkt. (je 2 Pkt.) Zeitkomplexität:

g(n) wird stets nur einmal berechnet

a) Berechnung von $g(n) = n \Rightarrow O(n)$

Bedingung: $i \leq n$ -> Schleife wird n-mal durchlaufen -> $O(n)$

Summenregel: $T(\mathrm{Pr}\,ogramm) \in Max(O(n), O(n)) = O(n)$

b) Berechnung von $g(n) = n \Rightarrow O(n^2)$

Bedingung: $i \leq n$ -> Schleife wird n-mal durchlaufen -> $O(n)$

Summenregel: $T(\mathrm{Pr}\,ogramm) \in Max(O(n^2), O(n)) = O(n^2)$

c) Berechnung von $g(n) = 2^n \Rightarrow O(n)$

Bedingung: $i \leq 2^n$ -> Schleife wird 2^n-mal durchlaufen -> $O(2^n)$

Summenregel: $T(\mathrm{Pr}\,ogramm) \in Max(O(n), O(2^n)) = O(2^n)$