

Poniżej znajduje się **opis funkcji matematycznych**, które można stosować w **formułach pytań obliczeniowych (Calculated / Simple calculated / Calculated multichoice)** w Moodle 4.5. Lista i zasady składni wynikają z dokumentacji Moodle dla typu pytania „Calculated”.

Źródło referencyjne: [Calculated question type – MoodleDocs](#)

1) Zasady zapisu formuł (ważne, zanim użyjesz funkcji)

1. **Nie wpisuj znaku = na początku formuły.**
 2. **Brak mnożenia domyślnego** – zawsze używaj * (np. $2 * \{P\}$, a nie $2\{P\}$ ani $2(\{P\})$).
 3. **Stosuj nawiasy** – szczególnie w funkcjach, aby uniknąć niejednoznaczności (np. $\sin(3/\{x\})$, a nie $\sin 3/\{x\}$).
 4. **Potęgowanie:** nie używaj $^$. Zamiast tego stosuj $\text{pow}(a,b)$.
 5. Funkcje mogą zawierać **wildcards** (zmienne) w $\{ \}$ oraz inne wyrażenia, np. $\text{round}(\{a\} * 1.23, 2)$.
-

2) Dostępne funkcje matematyczne w pytaniach obliczeniowych Moodle

Moodle dopuszcza więcej niż podstawowe operatory arytmetyczne. Poniżej grupuję funkcje według zastosowań (z krótkim opisem i typową składnią).

A) Wartości bezwzględne i zaokrąglenia

- $\text{abs}(x)$ – wartość bezwzględna.
- $\text{ceil}(x)$ – zaokrąglenie w górę do liczby całkowitej.
- $\text{floor}(x)$ – zaokrąglenie w dół do liczby całkowitej.
- $\text{round}(x, d)$ – zaokrąglenie x do d miejsc po przecinku (np. do 2 miejsc dla złotych/groszy).

Praktyka ekonomiczna (PLN):

| |
|--|
| 1 $\text{round}(\{kwota\} * (1 + \{vat\} / 100), 2)$ |
|--|

– typowy zapis „brutto” z 2 miejscami po przecinku.

B) Potęgowanie, pierwiastki i stałe

- `pow(a, b)` – potęga a^b (zamiast $^$).
- `sqrt(x)` – pierwiastek kwadratowy.
- `pi()` – wartość liczby π (jako funkcja bez argumentów).

Dokumentacja podkreśla, że **nie ma innych „stałych”** wprost, poza `pi()` (jako funkcją).

C) Funkcje trygonometryczne (uwaga: radiany!)

- `sin(x)`, `cos(x)`, `tan(x)` – funkcje trygonometryczne; **argumenty są w radianach**.
- `asin(x)`, `acos(x)`, `atan(x)` – funkcje odwrotne; wynik także w radianach.
- `atan2(y, x)` – arcus tangens z dwóch zmiennych; dokumentacja zwraca uwagę na kolejność argumentów jak (y, x).
- `deg2rad(x)` – stopnie \rightarrow radiany.
- `rad2deg(x)` – radiany \rightarrow stopnie.

Przykład:

```
1 sin(deg2rad({kat}))
```

Jeśli `{kat}` losujesz w stopniach, musisz przeliczyć na radiany.

D) Funkcje hiperboliczne

- `sinh(x)`, `cosh(x)`, `tanh(x)` – funkcje hiperboliczne (argumenty w radianach).
 - `asinh(x)`, `acosh(x)`, `atanh(x)` – odwrotne hiperboliczne.
-

E) Logarytmy i funkcje wykładnicze

- `log(x)` – logarytm naturalny (\ln).
- `log10(x)` – logarytm dziesiętny.
- `log1p(x)` – $\ln(1+x)$ liczone stabilniej numerycznie dla małych x.
- `exp(x)` – (e^x).
- `expm1(x)` – $\exp(x)-1$ liczone stabilniej numerycznie.

Przykład ekonomiczny (ciągłe oprocentowanie):

```
1 {P}*exp({r}/100*{t})
```

– wartość przyszła przy kapitalizacji ciągłej.

F) Minimum/maksimum i operacje „reszty”

- `min(a,b,...)` – minimum z wartości.
- `max(a,b,...)` – maksimum z wartości.
- `fmod(a, b)` – reszta z dzielenia zmiennoprzecinkowego (modulo).

Przykład (ekonomia): ograniczenie rabatu do przedziału 0-30:

```
1 min(max({rabat},0),30)
```

G) Losowość i liczby całkowite

- `rand()` – generuje losową liczbę całkowitą (dokładny sposób użycia zależy od implementacji; w praktyce do losowania w pytaniach i tak częściej używa się datasetów).

Dobra praktyka (Moodle): do losowania wartości wejściowych preferuj **datasety** (zestawy danych), bo są przewidywalne, kontrolowalne i powtarzalne w raportach, a `rand()` bywa trudniejszy do „opanowania” w ocenie i weryfikacji.

H) Konwersje między systemami liczbowymi (rzadziej potrzebne w ekonomii)

- `decbin(x)` / `bindec(x)` – dziesiętne ↔ binarne.
 - `decoct(x)` / `octdec(x)` – dziesiętne ↔ ósemkowe.
-

I) Funkcje sprawdzające „poprawność” liczby

- `is_finite(x)` – czy liczba jest skończona.
- `is_infinite(x)` – czy jest nieskończonością.
- `is_nan(x)` – czy „nie jest liczbą” (NaN).

Te funkcje są przydatne głównie w sytuacjach, gdy istnieje ryzyko dzielenia przez zero lub logarytmu z liczby nie-dodatniej (warto temu zapobiegać już na etapie doboru datasetów).

3) Krótka „ściąga”: typowe formuły ekonomiczne w Moodle

VAT + rabat (2 miejsca po przecinku)

```
1 round(( {N}*(1-{d}/100))*(1+{vat}/100), 2)
```

Zalecane: round(...,2) i konsekwentne ustawienia tolerancji/formatu odpowiedzi w pytaniu.

Procent składany (kapitalizacja roczna)

```
1 {P}*pow(1+{r}/100, {t})
```

Potęgowanie tylko przez pow.

Procent składany (kapitalizacja ciągła)

```
1 {P}*exp({r}/100*{t})
```

4) Bezpieczeństwo i dobre praktyki (Moodle 4.5)

1. **Preferuj datasey zamiast „losowości w formule”**. Datasets są kontrolowane i mniej podatne na błędy oceny/raportowania niż generowanie liczb w samej formule.
2. **Pilnuj dziedziny funkcji** (np. $\log(x)$ wymaga $x>0$, \sqrt{x} wymaga $x\geq 0$). Najlepiej zagwarantować to zakresem datasetów, zamiast „ratować się” wyjątkami.
3. **Trygonometria = radiany**. Jeśli w datasetach losujesz stopnie, zawsze stosuj deg2rad.
4. **Nie próbuj „dopisywać” własnych funkcji na produkcji**. Istnieją materiały deweloperskie o rozszerzaniu funkcji (EvalMath), ale same dokumenty deweloperskie mogą być historyczne/deprecated i jednocześnie zwracają uwagę, że część funkcji może być „unsafe” (niebezpieczna). Takie modyfikacje powinny być wykonywane wyłącznie przez administratorów/developerów i poprzedzone analizą ryzyka.